

**SEPEC Conference
PROCEEDINGS**

NCC 9-16

hypermedia & information reconstruction

Aerospace Applications & Research Directions

ADDENDUM Final Proceedings

December 3 - 5, 1990

Houston, Texas

South Shore Harbour Resort & Conference Center

Co-Sponsored by
**University of Houston-Clear Lake
NASA/Johnson Space Center
Hypermedia Working Group**

N91-2195b
--THRU--
N91-2195b
Unclass
0332471

(NASA-CR-187936) SEPEC CONFERENCE
PROCEEDINGS: HYPERMEDIA AND INFORMATION
RECONSTRUCTION. AEROSPACE APPLICATIONS AND
RESEARCH DIRECTIONS. ADDENDUM (Houston
Univ.) 30 p
CSCL 053 33/62



Keynote Address

Hypertext, Typographic Man, and the Notion of Literacy

Patricia Ann Carlson

**Center For Advanced Media, International Centers For Telecommunication Technology
Rose-Hulman Institute of Technology**

Patricia Carlson is currently a National Research Council Senior Associate at the Human Resources Laboratory, Intelligent Systems Branch, Brooks AFB, San Antonio, Texas, researching the integration of intelligent hypertext systems and neural networks. Her current areas of interest include hypertext as a form of knowledge representation and AI applications in the areas of knowledge management systems, intelligent tutoring systems, and virtual world technology.

Hypertext, Typographic Man, and the Notion of Literacy

by

Patricia A. Carlson

University Resident Researcher

Air Force Human Resources Laboratory

Intelligent Systems Branch

Brooks AFB, TX 78235-5601

Introduction

This is not a technical talk; however, it is a talk about technology. Specifically, I'm going to talk about hypertext, the coming of the Knowledge Age, and the Twilight of Print as we know it. In order to do this, I'll appropriate the perspective of (1) the cognitive psychologist, (2) the cultural anthropologist, (3) the historian of technology, (4) the information scientist, (5) the futurist, and (6) the educational reformer.

The story I'll tell is of epic proportions, covering millennia of human development. In some ways the tale I've come to tell you today is a sad one, because it predicts the loss of old friends. In others ways, it is "gee-whiz" speculation -- resembling a modern Tom Swift gadgetry romance: *better living through cybernetics!* In still other ways, the talk may seem like an echo chamber from the 60s, filled with reverberations of Marshall McLuhan, Ted Nelson, and other would-be prophets who preached the gospel of the global village; of computer liberation; of technologically-induced millennia.

Let these impressions fall as they may: let me tell you up-front that my real intention is to have the effect of a locker-room pep talk at half-time when the team is down by a substantial number of points. Specifically, I want to suggest that this conference, the interest you've displayed by coming, and the papers we'll hear in these two days are part of a profound movement in the intellectual history of humankind -- the beginnings a virtual age for information, when text is produced, transmitted, and consumed in electronic form.

Certainly anyone with blood running through his or her veins knows that some kind of change is in the wind. As knowledge workers, educators, administrators, or in general people committed to intellectual work and a life of the mind, we live in disturbing times. Statistics on traditional literacy are headed downward. Many people in third-world countries can't read or write, and the global economy makes it doubtful that resources will be committed to the kind of massive educational push it would take to remedy the situation. Perhaps more telling, blue-collar workers in developed nations (people who only a generation or two ago valued education as the key to upward mobility) seem no longer to practice the skills they have, nor to encourage the development of textual literacy in their children. These are distressing facts to consider, unless one posits that the demise of traditional literacy is the harbinger of another kind of literacy.

" . . . One Dying, the Other Powerless to be Born"

Almost 550 years ago Gutenberg's supposedly serendipitous connection between the mechanism of a wine press and the function of a signet ring produced movable type. The world was on the road to the Information Age. According to scholars such as Harold Innis, Marshall McLuhan, and Walter Ong, humankind gave up much in exchange for textuality. Preliterate cultures actively construct meaning from all the sensory stimuli and rely more extensively on intuition and impression over logic and analysis. Literate cultures, on the other hand, undergo a distortion in the ratio of the senses -- to use Marshall McLuhan's terms -- until sight comes to dominate all others and rational/logical ways of knowing become pathologically efficient.

Given the speed and volume of written communication today, we need to stop and remind ourselves that writing is a skill humans acquired only about 5000 years ago. Using a uniform symbol system to represent information seems to have been initiated for a purely utilitarian purpose (the Hittites were successful warriors and merchants in the ancient world and needed a way to record their plundering and their proliferating business transactions). In the early days of writing, books (clay tablets) were widely adopted because they were a durable storage medium not because they were a versatile representation of thought. Over time, various symbol systems -- complex and subtle enough both to portray information and to express thought -- came into use (as with the hieroglyphics of ancient Egyptian monuments).

Folklorists were the first to delve into the cultural effects of literacy as they studied the transmission of epic poems, sagas, and other artifacts of the oral tradition. Anthropologists elaborated on the concept by identifying major differences in attitudes toward knowledge in oral and literature societies -- which in turn determined such things as allocation of resources, professionalization of certain activities, and distribution of wealth. Cognitive psychologists added theories of how mentation and ways of knowing differ in pre-literate and literate cultures. Yet, too often the early studies reflected a bias. Either they took the view of the logical positivist and insisted on dichotomies (such as non-logical versus logical, "primitive" versus domestic) or they painted romanticized visions of an idyllic life shattered by technology and industrialization (such as quaint depictions of harmonious tribal life).

Ways of knowing are fundamentally different in oral and in literate societies.

- o **Process of Meaning:** Lacking a permanent artifact to record sequential thought, oral cultures are process-oriented rather than product-oriented. This process includes a dimension of participation or performance, as speaker and listener actively construct the dialogue. On the other hand, the phonetic alphabet and familiarity with print allow meaning to be stripped from the page without the intermediary of vocalization. The scribal tradition of reading aloud becomes pointless. The locale of text, previously the abbey scriptorium abuzz with mumbling, as monks mouthed the words they copied was replaced by the austere silence of a library.
- o **Patterns of Meaning:** Acquiring, maintaining and disseminating information in an oral culture require that forms be easy to remember. Reflecting this, narrative and description (the more "primitive" devices of rhetoric whose organizational features are derived from time and space) prevail over analysis and argumentation (patterns which require logical manipulations such as classification, comparison and contrast, and criteria for establishing hierarchical organization in bodies of information). Permanence in an oral culture is equivalent to remembrance, which is facilitated through storyline and mnemonic devices. Permanence in a literate culture is equivalent to acceptance, which is determined by how well the author formulates his or her thesis, develops the argument through evidence, and adheres the conventions of form.
- o **Provenance of Meaning:** Oral cultures have no concept of accuracy, of originality, or of plagiarism. Truth (here defined as the authority of the spoken word) exists as the outcome of the social making of meaning.

Literacy as we traditionally use the term means skill in reading and writing. This proficiency is commonly the product of several years of formal, specialized schooling and a tremendous commitment on the part of students and teachers to the value of being literate. However, our society is in the midst of a potentially dangerous split between those who participate in traditional literacy and those who either cannot or who chose not to. Let me review with you some observations on the state of literacy today. It does not much matter whose statistics we use; they are all cause for concern.

- o **Historical Dimension:** Though the focus on literacy promises to intensify in the 1990s

as nations compete more fiercely in the global market place, the severity of the problem was acknowledged in previous decades. For example, former Secretary of Education Terrel Bell testified before the House (in 1982) that "in 1975, 63 million Americans . . . [were not] proficient in meeting the educational requirements of everyday adult life." By 1982, Secretary Bell noted, this had risen to 72 million, which amounted to half the adult population. [37]

- o **Measurement and Assessment:** Nationalized tests tell a dismal story. The College Board's Scholastic Aptitude Tests demonstrate a virtually unbroken decline from 1963 - 1980. [25] While a slight improvement occurred in the mid-1980s, the scores have once again started to decline, even with all the attention these plunging figures have received.
- o **Misguided Focus:** Defining the problem as one of literacy versus illiteracy masks a significant problem. Many who are categorized as "literate" cannot practice these skills with a degree of proficiency needed to fully participate in a post-Industrial society, let alone a post-Information society. For example, according to the National Assessment of Educational Progress, nearly 40 percent of the 17 year-olds tested cannot draw inferences from written materials, and only one-fifth can write a persuasive essay. [25]
- o **Social Values:** Having an educated labor force is an economic mandate; having an educated electorate is the life blood of our democratic society. Therefore, in addition to the "literate," the "under-literate," and the "illiterate," we need to consider the "a-literate": persons of adequate abilities who have simply succumb to inertia. Reading and writing sophisticated prose on a regular basis; challenging and provocative textbooks in schools; a thriving print industry (including newspapers dedicated to the professionalism of the press as an institute), and thriving bookstores and public libraries: all are dwindling as the country increasingly bifurcates into educated elite and anti-intellectual masses.

* * *

I could fill all my time here today documenting the decline of traditional literacy. Instead, I'll just draw the obvious conclusion and move on: Current ways of teaching the cognitive skills necessary for life in a complex society are failing a significant portion of the populace. For Marshall McLuhan, the widespread indications of the breakdown of print technology would be the expected end-result of centuries of progressively removing knowledge from the well-springs of its birth in an integrated sensorium. Not only are such failures predictable for McLuhan, they are welcome because they hasten the change toward better ways of doing things. Yet for most of us, living in what appears to be the end of traditional literacy and the onset of a new dark ages, the transition may be more like the intellectual twilight described in Matthew Arnold's "Dover Beach." We seem to be caught in the murkiness between one dying age and another age, as yet undefined and seemingly powerless to be born.

The Technology of Text: The Tyranny of Text

Like most advocates of hypertext, I'll begin by giving the case against linear text. This indictment is usually made by considering the limitations of paper as a medium. The standard arguments go something like this:

- o **Limited Access:** Paper text (or flat text) provides only two dimensions of information: linear or hierarchical, while hypertext more closely models the deep structure of human idea processing by creating a network of nodes and links, allowing for multi-dimensional navigation through a body of data.
- o **Rigid Structure:** A second set of complaints stems from the brittleness of printed

materials -- the inability or the reluctance to decompose the book either logically or physically. Books are structured representations of a knowledge domain. Their strength is that they give order and permanence to information; their weaknesses derive from these same features. The organizational schema may not map well with the immediate needs of a specific reader. The definitive aura of text is intimidating to the novice. And, the static nature of printed material doesn't easily allow the dynamic reorganizations that a domain expert frequently requires.

My comments are more anthropological in that I want to consider the book as an artifact of the mind which emerged out of invention, developed over time, and will be displaced when worn out. This is predictable: the history of our species is one of competition and displacement, and nowhere is this more evident than in the development of the human intellect. For half a millennium, printed text served as a platform for the storage, access, structure, and compression of information. But times and requirements have changed. We've lost both the time and the technique for effective communication with these passive artifacts.

To chronicle the rise and the decline of the book as an instrument of the mind, I'll recount the work of two communicologists: Elizabeth Eisenstein (an historian by training) and Marshall McLuhan (a pop-culture hero by acclamation). Both writers would agree that technology is not a neutral force in human development. Both contend that, just as written communication had changed social organizations in observable ways, so the reliance on print technology over time has changed the psychological processes of the individual. Eisenstein covers the early history of print technology, when the book was a new plaything, a vibrant and powerful tool for shaping knowledge and enhancing habits of mind. The more caustic comments of McLuhan come from his focus on text in the twentieth century -- at a time when the medium is being challenged by robust forms of electronic communication.

In her major work, The Printing Press as an Agent of Change: Communications and Cultural Transformation in Early-Modern Europe (1979), Eisenstein offers a balanced, well-documented discussion of the political and social consequences of literacy. Her arguments center on three issues.

- o **Information Dissemination:** Though the image is one of scattering and dispersal, Eisenstein indicates that the wide dissemination of books made possible through printing caused something of the opposite to happen in that ideas came together in repositories -- either bookshops or libraries. Additionally, being well-versed on a topic no longer required a wandering scholar who had visited the various locations containing the precious manuscript copies of significant texts. Scholars, especially scholars with financial backing, could stay in one place and have the information come to them. The new distribution put an end to the *glossator* (or commentator) and the intellectual activity of spending a lifetime layering interpretations on a single text. Instead, new habits of mind developed based on new intellectual permutations and combinations. Well-stocked bookshelves led directly to comparisons, contradictions, indexing, and referencing.
- o **Standardization:** As a second major observation, Eisenstein points out that the claim that printing produced standardized texts has been overstated. To the contrary, the rush to capitalize on the new markets for books carried with it slipshod workmanship and entrepreneurial editors who had little concept of modern-day techniques and standards. Historians of early print can demonstrate that texts from this early phase of printing were even more prone to error than were the laboriously handcopied manuscripts of the previous age. Only when public pressure came to bear -- in the form of economic sanctions -- did the concept of correctness in the printed word take hold.

More central to this discussion is another form of standardization for which the first mass medium was immanently suited: print as the purveyor of fashion. Pattern books made more uniform the design of virtually every social artifact, from clothing, to furniture, to

calligraphy. And just as these objects were reduced to a sameness while simultaneously being elevated in quality, so forms of thought -- such as mass opinion, public education, and the spirit of nationalism -- were simultaneously improved, amplified, and homogenized. Primers, catechisms, devotional tracts, and books of statecraft and history took on an orderly, arranged sameness, both of appearance and content, as more laymen came to look to the printed word for clear cut answers to questions which previously would either not have been asked or would have been subjected to conversation, debate, dialogue. As the ultimate example of this trend, the ambitions of the growing middle class and regularization met in the common, popular handbooks for achieving worldly success. Perhaps the most (in)famous of these ways-to-wealth is Machiavelli's "career cookbook," The Prince.

- o Methods of the Mind: The widespread dissemination of books noted above also induced new methods of dealing with information. Although these skills and the level at which they were applied seem elementary to today's knowledge worker, they were either unknown before the Gutenberg press or were not used with any degree of precision by large numbers of the population.

For example, publishing increased the urge to collect data, to rationalize format, and to codify bodies of knowledge. Therefore, editorial decisions had to be made about structuring and accessing information in reference works. Existing models were few and relatively impoverished. Medieval manuscripts were usually compendia containing a variety of texts bound together more by the circumstances of chance and convenience than for any reasons of commonality. While owners of such collections frequently constructed indexes, they were individualized and highly idiosyncratic. Annotations in margins (diagrams, brackets, cross-references, terse comments) abounded. Occasionally, some enterprising scholar even marked sections with tabs. But these devices amounted to personal mnemonics rather than consistent cataloging. Prior to the sixteenth century there is nothing of the orderly systems one associates with a culture grown used to handling and storing information, until printers -- driven by competition to design more useful products -- hit upon the concept of using alphabetical order.

Organizing bodies of information into predictable patterns is but one instance of a cluster of skills induced by the improved "information science" of bookmaking. Documenting her position through extensive research, Eisenstein argues that the conventions governing the use and transmission of information -- expectations that we take for granted today -- are a direct result of print technology. Perhaps more telling, Eisenstein argues that decades of participation in the culture of print have transmuted into individual cognitive processes to the point that alphabetic literacy dictates how we define ourselves and the world around us. Consider the following observations as an indication of just how much of western intellectual tradition has been shaped by the printing press.

- *Requirements of Rationalism*: Print acquainted the reading public with the modes of inquiry that became the underpinning of the western tradition in science: (1) comparison and verification of data, (2) context-independent forms of reasoning, and (3) the accumulative tradition in research.
- *Requisites of Form* -- Some of what we think of as "modern prose" is an accommodation for print conventions and the physical process of placing text on a page. Other aspects reflect ways of thinking and the socioeconomic changes engendered by print itself. For example, specialized uses of language for the professions (law, business, applied sciences) lead to formalized texts exhibiting the types of unity, coherence, organization, and development amenable to the principles of the particular profession.
- *Power of Authority*: In the classical oral tradition, oratory and debate were respected means of argument and persuasion. Text, of course, removed the dynamic

exchange factor. To convince, authors needed to anticipate the responses of the reader, to provide either honest answers or workable subterfuges, and -- in general -- to build tightly-knit, reasonable structures. This strong tendency toward definitiveness in writing combined with the very permanence of print to create the concepts of authorship, of intellectual property, and of ownership of ideas (and the reverse, the notion of plagiarism).

To summarize then, Eisenstein sees the printing press as having had a profound effect on the development of the human mind and spirit. In her two volume set, she argues that cognitive and cultural changes brought on by the technology of text made possible the scientific revolution and the Protestant Reformation.

Turning to Marshall McLuhan, we hear a less favorable interpretation of the development of typographic man. Earlier in my talk, I spoke about **the problem of literacy** in our increasingly-complex world. For McLuhan and others who have echoed his position [33] **literacy is the problem**. In McLuhan's view, the sequential organization of information characteristic of printed documents is equated with the rational presentation of information. McLuhan's typographic man is the passive recipient of prefabricated "method of mind" or standardized patterns for perceiving reality. Once textbooks became the predominant "teaching machine" of the classroom, there was nothing to stop the standardization of thought and conscious. The improved organization of information made possible by movable type and many decades of overemphasis on sequential, uniform, and lineal thought culminated in itemization, and classification as the staples of learning. McLuhan sneeringly characterizes the paradigm of knowing for typographic man as " . . . proceeded by cold-blooded definitions and divisions leading to still further definitions and more divisions, until every last particle of the subject has been dissected and disposed of." [28]

Toward the end of his life, McLuhan (who died in 1980) found a compelling metaphor for his ideas in the notion of brain hemisphericity. His characterization of the right/left brain dichotomy is pretty much standard fare for today's audience, which has been inundated with popularized versions of the bicameral mind for the past decade. But for McLuhan, the emergence of biological explanations for different ways of processing information based on hemispherical dominance was a welcomed corollary to his own work. Typographic man specializes in left brain functions -- linear and analytical. Oral cultures (and to some extent those cultures whose writing system is not based on an alphabet) specialize in right brain functions -- simultaneous and holistic.

Though McLuhan is a harsh critic of typographic man's limitations, he did not intend to ally himself with doomsayers. He was trying to startle us out of our smugness and complacency by showing that information technologies need to be liberated from the chock-hold of rationalism if the human spirit is to move on to its next stage of development. In fact, The Gutenberg Galaxy, and its sequel, Understanding Media, are intensely optimistic: both exuberantly forecast a millennia for humanity. Additionally, even though McLuhan harkens back to a "unified sensorium" -- a sort of golden age where information (and hence knowledge and wisdom) are fabricated in the individual consciousness through multiple sensory stimuli -- he cannot be charged with antiquarianism. To the contrary, he embraces modern technology and its potential to extend human awareness as the mechanism for the new awakening. The second age of integration will not be one of *orality* but one of electronically-induced *simultaneity*.

Hypertext and the New Literacy

Let me start with the basics. The elements of hypertext are nodes (chunks of text or graphics) and links (connections indicating a relationship). Together, they give hypertext the ability to fashion knowledge from information. Nodes deconstruct the linear sequence of printed materials, making possible dynamic recombinations. Links not only connect nodes but can also carry with them

information about the connection, turning text into a web of relationships.

But hypertext -- and its extension hypermedia -- are certainly more than this simple definition. The progress of Gutenberg's print technology as a vehicle for literacy is one of delimitation: it is a scenario of intensifying concentration and focus, of improving proficiency by taking things away: sound, touch, picture, movement: the simultaneous interplay of sources which create an amalgam of awareness, perhaps a collage, maybe even a kaleidoscope, whose construction did not necessarily have to stand up to the rigors of rational, sequential patterns of thought.

The image of hyper-technologies as a vehicle of literacy is one of progressively incorporating all the faculties and returning to the premise that meaning is a multi-sensory product. Nodes can be text, computer graphics, computer code, sound clips, still or motion pictures, animation, or combinations. Modalities (as alternative vehicles of knowing) lost to typographic man -- sound, touch, movement, and vision -- return with enhanced vigor.

Already the idiom of information processing has begun to change. Characteristics begun in hypertext and extended in hypermedia will increase in density and complexity with future development. Even today, the language used to describe the hypertext/media interface is steeped in visuo-spatial-tactile implications. Collections of information become "pools" and users "navigate" through them. Content becomes "knowledge space" and users take "tours" or make their own "path" while "walking" through the structures. These metaphors of interaction become more immediate as impersonal *information* is transformed into intimate *environment*.

Synesthesia (describing one sensory experience in terms of another) abounds in describing the conventions of these hyper-worlds. Input/output takes place through a variety of devices, from the standard (keyboard, mouse, and joystick) to the more exotic (voice activation and data glove). Even higher-level cognitive processes now have a tactile/visceral referent, allowing educators to speak in terms of "muscle memory" for mentation that used to be considered so abstract that it had no descriptor. Again, new terminology abounds. Controlling a sequence of complex mathematical computations to change locations in a database by rotating your hand inside a data glove now becomes a "gestural utterance." [15] Layers of information intermediaries have been removed, and we are given a more immediate interface to thought.]

* * *

I want to be careful, however, to make clear that, in my opinion, the new literacy will be text-based. Other literacies, based on other technologies, exist today. Computer literacy, while not now practiced by the masses, is a robust field, supported by professional practitioners and theory-producing scholarship. The same can be said for mathematics. As for film -- one can study and practice the celluloid idiom on expert levels. This is true for sound (such as music), for movement (such as drama or dance). In other words, there are as many literacies today as there are symbol systems. This fragmentation, the building of barriers, and the limited understanding between modes of expression may be the negative legacy of print McLuhan was talking about.

Let me repeat: I am not here to forecast that text literacy will be replaced by any of the current alternative literacies. I doubt seriously that, having spent 500 years becoming typographic man and 5000 years becoming text literate, that humankind will abandon this modality anytime soon. The sheer bulk of knowledge lock up in paper text makes it unlikely. To abandon these artifacts now would be a "book burning" unlike any known to history. However, this does not mean that books as we know them are effective artifacts of the mind. Nor does it mean that the medium of print, which taught humankind so much during its inception and growth, may not have simply worn out over time. We can abandon the medium while still clinging to the modality.

My claim is that hypertext, at its root, enriches many of our current, print-derived information conventions and draws renewed strength and vigor by incorporating cognitive activities from other forms of literacy. More to the point, hypertext as a **hybrid** may reunite typographic man with

tribal man; may mediate between right-brain and left brain; may merge intuition and immediacy with rationalism and analysis. The new amalgam, this hyper-literacy, is characterized by four keywords, each a cluster of powerful strategies and heuristics for manipulating information and experience.

- o **Association:** One widely-accepted model for human memory posits that entities ("chunks" of information) are stored in human memory as items linked in a pattern of associations. Like a *gestalt*, this pattern of nodes and links becomes more than the sum of its parts. Unlike flat databases access, search in the human mind addresses the clusters formed by relationships. This ability may account for such typically human ways of knowing as intuition, following a hunch, and flashes of insight.
- o **Abstraction:** Because humans are capable of forming categories from observed instances, the landscape of our mind is not a jumble of discrete properties and features. The ability to perform feature-extraction analysis (through a mechanism similar to comparison/contrast) produces higher-order representations. Individuals "learn" by augmenting, combining, and rearranging a layered collection of cognitive maps, many of which are overlapping and most of which are interconnected through a complex network.
- o **Plasticity:** In much of human cognition, the process behind the product is abstract -- leading to a kind of opaqueness which keeps us from formulating the strategies necessary for expert behavior. What we need are tools that help us to visualize a complex problem space. Transforming the unobservable into objects that can be examined and inspected provides a mental model where ideas can be represented and related in different configurations and on different levels of specificity.
- o **Performance:** Direct manipulation -- a relatively new idea in human-machine interface design -- was a standard feature of information processing for oral cultures. Additionally, "acting-out" complex values and beliefs through ritualized performance was common practice. This craving to engage hand, eye, and brain in cognitive activities remains with us, attesting to the powers of motor feedback even in highly sophisticated forms of mentation.

When fully realized over time, this hyper-literacy will increase our powers to *conceptualize*, enhance our ability to *communicate*, and augment our capacity for *creativity*.

But what might these new "artifacts" of the mind be like? We know what books are like because they are concrete, they've been around for about 500 years, and their design has remained relatively stable during the last few centuries. The artifact of the new literacy will be virtual and, therefore, best described by analogy and metaphor. Somewhat in a sportive vein, but also equally serious, let me suggest two defining analogies for *future-text*.

(Hyper)Text as Peripheral Brain

Though the idea of "peripheral brains" may at first seem contrived, they actually do exist. Look around you and you will see many. And these peripheral brains are rich repositories of information, they're user-friendly (unless, of course, they're having a bad day), they can adapt content, level, and style of response. They can handle fuzzy logic and incomplete queries. In fact, they can frequently guess your needs based on subtle visual and verbal cueing.

Now imagine you're in a scene from Star Trek -- you walk into a large, cavernous room, filled with row upon row of shelves. And on these shelves are brains -- that's right, disembodied brains sealed in amber and incapable of communication. Now the encased brains begin to blur and fade slowly into books, and you realize that you are in a library.

The first wave of hypertext concentrated on the nodes and envisioned hypertext as an enhancement

to online information delivery systems. Second generation hypertext focuses on the linkages and exploits the enormous diagnostic and tutoring potential of the web (which is a form of knowledge representation as powerful as the if-then rules of an expert system, and certainly more flexible and synoptic than production rules). In other words, second generation hypertext suggests that the web is a new AI formalism. As a much softer, more tractable form of knowledge representation than many of the AI methods currently in practice, second-generation hypertext may make intelligent systems much more of a reality to the masses than was previously possible.

Like a good physics problem, I'm going to assume away all those pesky details of reality that get in the way of solving the problem. So, I'm not going to worry much about how we convert existing text to hypertext or the enormous task of designing acceptable interfaces to the threaded result of this conversion. Rescuing or retrofitting existing text is, of course, a tremendous technical challenge. Nothing now on the market has all the features we'll eventually want for turning text into enriched, usable hypertext. Yet, I know of several such "Shread-N-Thread" products already being coded. Their approach, in essence, assumes that written text is but one form of knowledge representation. Rhetoricians and discourse analysts can demonstrate that writing has predictable patterns, although the variety and the variation makes these structures at times hard to detect and represent. These difficulties aside, it would seem that ultimately, we will want to look for equivalences in our various forms of knowledge representation and, more importantly, to automate the translation process. (See Figure 1).

But let's assume that the automation has taken place and what we have is a richly interlaced body of information. Implementing this metaphor of text as brain changes the role of the human in the loop and supports new ways of extracting knowledge from text. (See Figure 2.) In the traditional reading process, the human provides the interpretation by internalizing a flexible representation of the text's infrastructure. However, only experienced readers are proficient at meaningfully decomposing the monolithic structure of text. For example, expert readers can vary speeds (skim, scan, thumb) to extract meaning. They also have a wider repertoire for manipulating the design features of the artifact (estimating a knowledge "footprint" from format and layout cues, fast lookup routines, integration of text and graphics). In a hypertext system, a good portion of this expert behavior either has been automated or the need for it has been eliminated. But the gain is more than just convenience. The new medium encourages information processing behaviors that were not possible in flat text. Two of these (generically defined as **exploration** and **consultation**) are given more attention below.

- o **(Hyper)text as Database:** In unstructured hypertext, the reader is presented with a densely interlaced collection of nodes. Links interconnect occurrences of the same idea and perhaps related concepts. Readers follow links based on "need to know" or on curiosity. Blazing associational trails through a rich body of information can be fun, as anyone who has ever lost an hour or two browsing the encyclopedia can attest. The down side of such browsing is that the reader, even one who begins in a goal-directed mode, may start out reading about Admiral Chester Nimitz and end up perusing the history of ballet.

While this "hunter-gatherer" style of traversing webs has its attractions, the kind of access behavior I'm envisioning here is almost the opposite of this non-directed intellectual grazing across a field of information. In fact -- to continue the image of knowledge as terrain -- the kind of behavior I'm thinking about would be more like geographically surveying the textbase to find not only the length and breath of the field but to map the contours (the peaks and valleys) of the domain. Such knowledge probing could be implemented in a number of ways, once text becomes much like an entity-relationship database.

Focusing on the information embedded in the hyperweb allows for concept searching rather than content searching. Dependent upon the application, the user may perceive that a

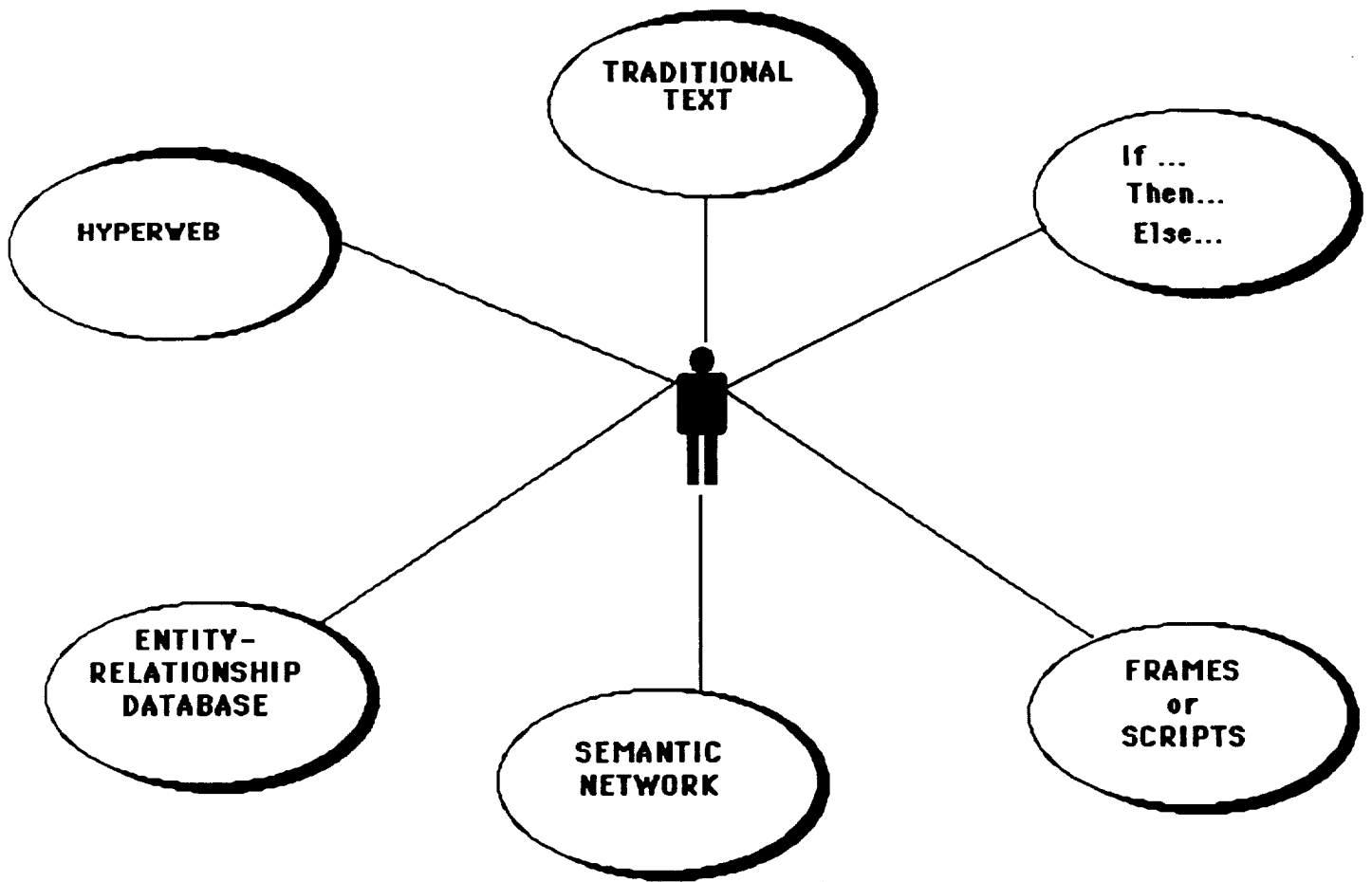


Figure 1: Varieties of Knowledge Representation

Text and Traditional Literacy



HyperText and the New Literacy

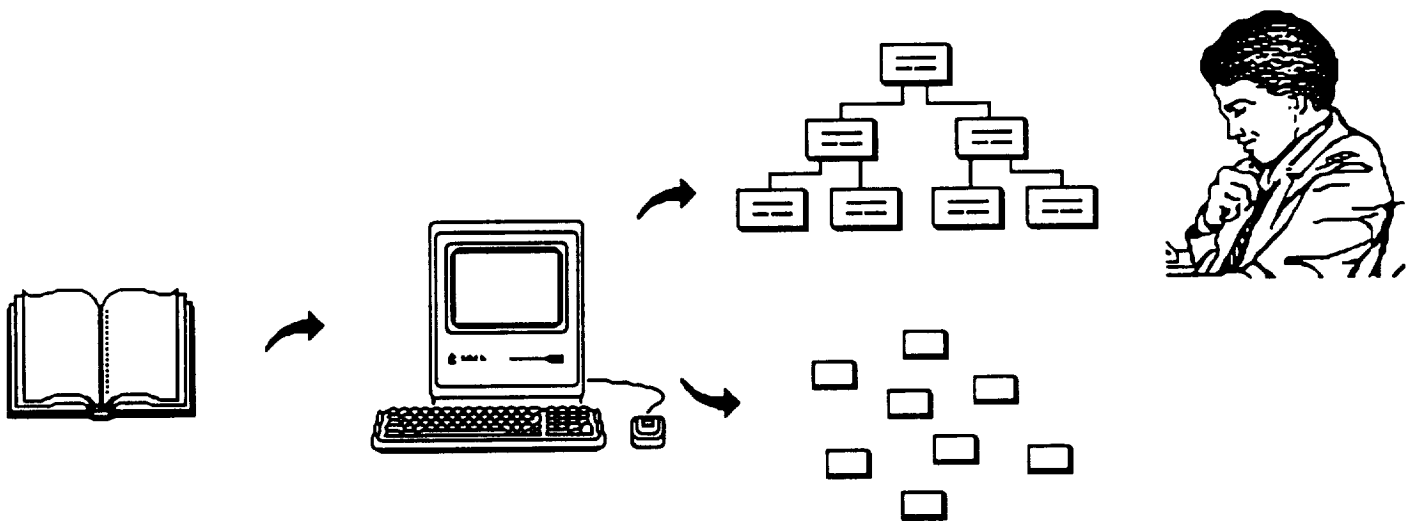


Figure 2: The Role of the Human in the "Reading" Process

concept can be represented by a specific cluster of nodes and/or link types and attributes (As a simple-case example, in a genealogy, asking for the earliest, male node with the fewest number of son of daughter links would probably produce the name of the first confirmed bachelor in the family.)

We are now only at the threshold of what concept searching can do for the growth of human knowledge. Imagine, for example, being interested in the social impact of the defeat of the Armada and being able to sit down to the online version of the Oxford English Dictionary and ask for an alphabetical listing of all Spanish-root words entering the English language between 1688 and 1690. And then being able to read the earliest extant use of the word in print and getting the bibliographic citation for that document.

Most of our knowledge structures have been designed for the convenience of their creators and curators, not for their users. For example, in using a library, how much of your energy is tied up in manipulating the physical limitations of achieved information as opposed to coming to terms with the logical and abstract dimensions of the knowledge. A test-wise student knows how to reverse-engineer a reading comprehension test: it is far more efficient to look at the list of questions before reading the passage. Isn't it time we all learned to reverse-engineer our antiquated, inadequate information processing strategies.

Imagine being a high school student of the future writing a term paper. A global, electronic library (such as envisioned by Ted Nelson) puts information at your fingertips much in the way that a utility company pipes energy into your home. You call up the interface and fill in a form, giving the criteria of the nodes you are seeking (initially, this will be keywords and scope, but in more advanced systems this profile might encompass such things as logic structures, semantic and syntactic characteristics, and complexity of graphics). A sophisticated retrieval mechanisms (combining both AI and standard information retrieval algorithms) returns a listing of items, rank-ordered based on the number of criteria satisfied by the node. You start with what appears to be the best choice for finding quality information and -- once inside the textbase web -- begin to follow the links indicated by highlighted terms.

At the opposite end of the spectrum from simple string searching, this kind of database probing requires the reader to conceptualize, to draw inferences about characteristics of bodies of information in a specific domain, and to work at a high level of abstraction. Additionally, the reader experiences -- first hand and almost immediately -- the power of these cognitive behaviors. Even the weakest reader can participate, and with computer-mediated feedback, can be shown how to model the inquiry so as to get useful results. What may start out as a trial-and-error process soon becomes a heuristic -- an insightful inference about the knowledge structure of a domain.

- o (Hyper)Text as Active Agent: Certainly in the world of computers, the idea of *interactive* has grown dramatically from the simple case of being able to sit down at a keyboard and display device and enter data rather than submitting a card deck. And, with increasing ability to add expertise on the computer-side of the exchange, human-machine interaction has taken on some of the characteristics of a dialogue. Ranging from diagnostic error-messages to programming shells which guide the human through complex manipulations, many examples could be cited of how sophisticated this notion of "interactive computer" has become. Perhaps the fruition of giving the computer a voice is embodied (literally) in Apple Computer's concept of the "knowledge navigator," a kind of personified filter which acts as a guide, an alter-ego, a partner, a counselor, an associate for the human user.

Once we start thinking of electronic text as a web of knowledge -- rather than a collection of bits and bytes -- all kinds of symbiotic relationships between words and reader are

possible. What I'm thinking about here is a system where text models "expert behavior" of performance (in a specific domain) through a guided-inductive approach that simultaneously empowers and encourages the user.

Imagine the potential for a knowledge domain to be represented in three-dimensional space. You're a civil engineer, working on a major project: an assessment of the water-treatment system for a medium-sized, mid-western town. Along with a team of support personnel, you've done extensive investigation and have collected numerous data and observations on the existing plant and such social factors as demographic and economic trends. All the reports have been entered into a central textbase. Now comes the real work: pulling this plethora of detail into a coherent and persuasive final report, with honest recommendations based on supportable inferences from the investigative phase of the project.

You go to your computer terminal and enter a program called *The Arena*. Your high-resolution screen presents a slowly rotating, multi-faceted shape, representing the aggregate of information you've collected. (See Figure 3.) You're asked to supply labels for as many of the out-lying circles as you need. Each represents a specific "view" of the database. What labels you use will depend upon the purpose of your report.

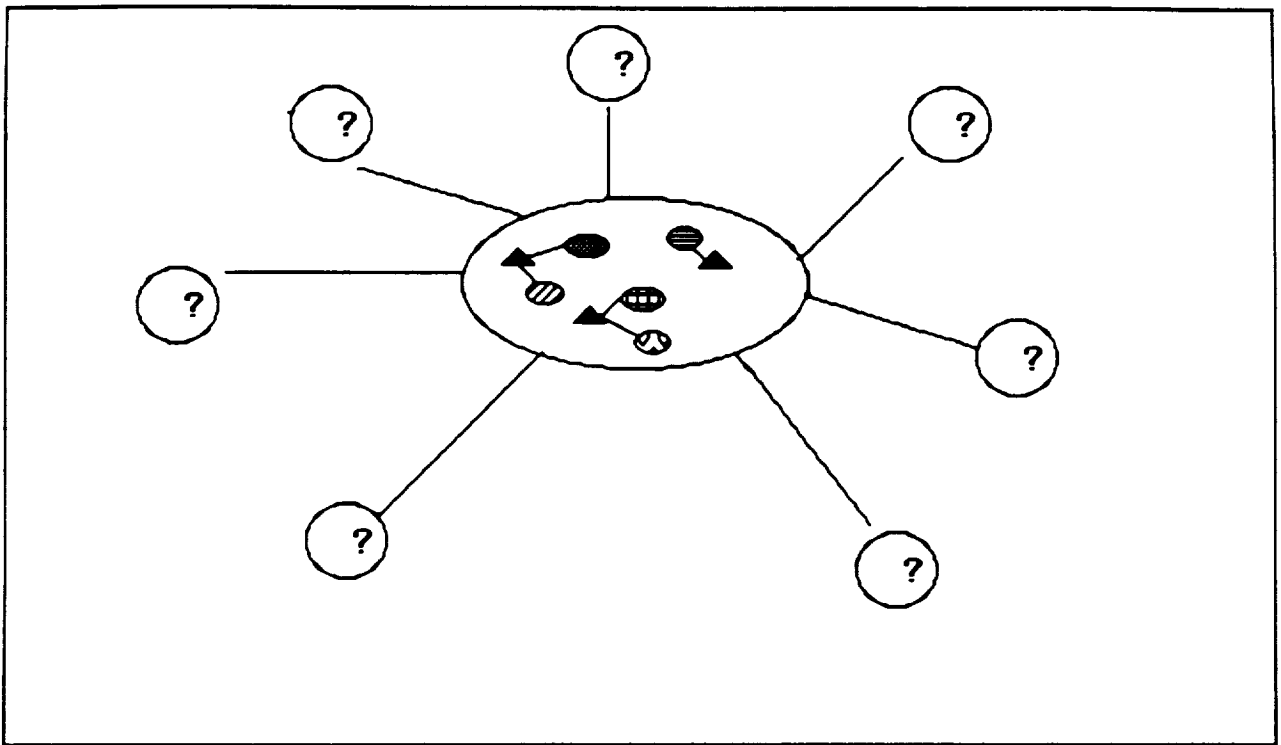
For example, if you are writing an advocacy package for replacing the old plant with modern technology, you might want to label the windows to represent the various constituencies either involved with or affected by the decision: city manager, demographic categories in the population (young/old, income, location, occupation), plant personnel, and the like. If you're certain the old plant will be replaced and your purpose is to compare and contrast two different designs for the new facility, you would use different labels for the windows.

These windows constitute the major categories of your inquiry. Next, for each window you will be asked a series of questions whose purpose is to elicit a set of keywords to create a more exact "profile" of the *view* from a specific window. Based on the criteria collected at this point, the system generates a hyperweb representing each view, perhaps even automatically labeling the relationship indicated by the links and the nature of the content of each node in the web.

Having this pre-threaded structure, you can now rapidly review large bodies of information, considering not only what items may be included in the final report but also the relationship among them. In the traditional writing process huge chunks of time are dedicated to this early exploratory work and it may be a matter of several days (or weeks) before a tentative outline emerges which adequately interprets text data into higher-level abstractions. Though not all the links in these prefabricated networks will be useful or even meaningful, the web becomes a place to start. Additionally, hidden relationships and significant cross references that might otherwise never surface may be discovered through this process. For example, if the network for each special-interest group in the water-treatment project could be superimposed on one another, the junctures and disjunctures could be read as areas of agreement and disagreement.

Since webs quickly become complex and their interconnects look like spaghetti when represented on the screen, a whole new category of tools -- called graphical interpreters -- could be developed. The semantically meaningful webs could then be passed through various filters to capture certain patterns and display the relationships in more easily interpreted patterns. For example, matrix analysis, state-transition diagrams, flow charts, or other quasi-formalisms could be used to bring intellectual control over the unwieldy body of information.

As with all the hypertext illustrations used in this discussion, the gain is more than just convenience. By reducing the initial need for time-consuming, low-level processing in a problem space, a hypertext system (such as the hypothetical *Arena* described above)



Adapted from Fairchild and Wexelblat, "Navigation through Cyberspace," MCC STP-161-89.

Figure 3: Interface for the "Arena"

changes the level of inquiry in complex, multi-dimensional tasks. By reducing the tedium, the system: (1) encourages the learner to deal with competing explanations of events, (2) fosters an appreciation for multiple points of view, and (3) empowers the learner to see phenomena as causally interconnected.

(Hyper) Text as Cognitive Tool

Moving on to the second serious parody I'll suggest for future-text, a quick sampling of the history of human technology suggests that "tools" fall into two functional categories:

- o **Extensions of Muscle:** Examples are easy to enumerate and generally involve harnessing external forces, such as draft animals, the steam engine, or nuclear power (for example, the forged plow, the wagon, windmills and water wheels, most forms of weaponry).
- o **Extensions of the Senses:** Examples can be characterized as some form of instrumentation which augments the human ability to perceive, measure, or manipulate (for example, the telescope, navigational devices, and surgical equipment).

A distinction should be noted here. Not all technology constitutes a "tool" in the true sense. The invention and eventual refinement of the chimney, for example, immensely benefited the construction of habitats, advanced the art of cooking, and added to human comfort. However, the chimney cannot be considered as a tool.

Gavriel Solomon -- of the School of Education at the University of Arizona - - gives an expanded meaning to the term "cognitive tools" by positing a category of cultural artifacts which -- by their very use -- shape our mental operations. [36] The argument that intelligence is not an indigenous quality of the mind but a product of structures of the mind and "technologies of the intellect" [20] has been made many times [33] [30].

Solomon explains that like most tools, the cognitive tool has a physical dimension. The difference is that a cognitive tool serves as an analog for a series of well-orchestrated, potentially complicated procedures, as in the process of calculating using an abacus. This distinct spatial form and hand-eye manipulations that accompany its use allow the user to internalize a representation of the tool (something akin to a mental model).

In considering the computer, Solomon posits a whole class of teaching programs, software which models new habits of mind for the user. He characterizes this learning exchange as "AI in reverse." Though Solomon does not mention hypertext as an example of AI in reverse, it is exactly this power to teach a mental model that will make hypertext the foundation of our new literacy. Two vehicles for inculcating mental models are discussed below. The first, **simulation**, is a seasoned concept, much used in current methods of training. The second, **scientific visualization**, is an old idea (the visual representation of quantitative data [41]) reconstituted through new technologies (graphical user interfaces, data-driven graphics, virtual reality).

- o **(Hyper)Text as Process Simulation:** It is in the realm of problem-solving that hypertext may have its greatest impact. In examining the process used by expert problem-solvers, cognitive psychologists have found certain consistent behaviors. Given a particular problem, experts have rich representations of what is asked for in the task, of the potential solution(s), and of the candidate methods for moving from one state to another in the process. Novices, on the other hand, appear to stall out and lapse into one of two dysfunctional patterns. Either they have an impoverished notion of what's expected, are easily satisfied by simplistic results, and truncate the process far before the finish (thus adding little to their cognitive repertoire). Or, they have no plan of operation, are

overwhelmed by fantasized constraints, and give up without meaningful results. Hypertext systems that model a mental process (decision analysis, design, instruction) provide the scaffolding needed for the novice to emulate the patterns of the expert.

For example, experts are proficient at deconstruction -- that is, partitioning the task into elemental components. The expert works on the pieces for a time, steps back to compare interim results with higher-level goals, consolidates gains, jettisons unrealistic expectations or excess constraints, re-orders plans (this might include satisficing), and moves back to working on the pieces again. The cycle takes place over and over during the problem-solving session. The expert excels where the novice fails because of this flexibility, this capacity to move smoothly between top-down and bottom-up strategies.

In writing, just as in software engineering, requirements must be worked downward through layers of specificity, until the outcome is an intelligible, encoded artifact. Both the University of North Carolina-Chapel Hill and Carnegie-Mellon University have large hypertext projects working on a computer-mediated environment to ease the cognitive process of writing. Though somewhat different, both have the same general characteristics. They both are based on a carefully researched cognitive framework for writing, and both recognize that significantly different patterns of mentation take place during the different phases of composition, and that, like the general model for problem solving, these phases are not accomplished linearly, but are cyclical.

The Writing Environment Project (UNC-CH) builds upon the theory that human memory is a semantic network and that during the formulative stages of writing -- as with any creative activity -- much of what is produced is a collection of associations. (The variety of doodles, sketchy outlines, disconnected notes, and various other flotsam and jetsam of discovery that most people turn out in the early stages of writing seems consistent with this model. First we churn through our associational web looking for relevancy.) At the present time, cultural conventions prevent us from thinking of this web structure as a sharable, informationally acceptable, formal artifact. So we must rework the connections into rhetorically meaningful patterns -- such as logical paragraphs with syntactically correct sentences -- that can be ingested linearly.

In much of education, the process behind the product is abstract -- leading to a sense of obscurity or opaqueness which keeps the learner from formulating the strategies that are necessary for expert behavior in the domain. To *reify* a concept is to transform the unobservable into objects that can be examined and inspected. The power of both the CMU and the UNC projects accrues from their use of hypertext as a reification device. In both cases a workstation-quality terminal serves as an automated storyboard -- a sophisticated variation of the notecard-yarn-and-thumbtack Rube Goldberg version of a knowledge processor -- where ideas can be represented and related in different configurations and on different levels of specificity.

The CMU project uses a matrix representation to help the writer compose an argumentative paper based on source readings (a variation on an assignment most of us remember as a term paper). The writer has the full-text of the reading sources online. She then fills in a matrix, placing the source authors' names on the vertical axis and the major issues on the horizontal. She then fills in the cells by using keywords and phrases which are anchored to the segments in the text where the point is made by the source author. The very process of filling out the matrix helps the writer to chunk the many issues and ideas presented in the separate sources into meaningful arrays.

Convenience is certainly a strong selling point for this system; however, something far more profound is occurring: the more obscure aspects of the writing process are *simulated* and the designer tests out strategies to manage the cognitive load of complex problem spaces. It is very likely that such heuristics, even those learned in simple forms, will transfer to

other more demanding situations.

- o (Hyper)Text as Visible Knowledge: The proliferation of text exacerbates the inadequacies of print. Living as we do in an age when the accumulation of the world's information doubles almost every 2.5 - 3 years, the narrow bandwidth of print is no longer adequate. Information compression through graphics has been advanced as a way of handling the load. Neurophysiology would indicate that this is a useful approach: about one-half the human neocortex is devoted to visual information processing. We have all grown up in a highly visual, auditory, and tactile, three-dimensional world and have amazing skills for processing billions of bits of information per second when encompassed in these environmental cues.

Hypermedia incorporates sight and sound into the web. Such technologies as CD-ROM, CD/I (Compact Disk/Interactive), and DVI (Digital Video Interactive) make it possible to include nodes which produce sound (voice, music, the world's longest belch -- as planned for the new Guinness Book of World Records), or display images (computer-generated graphics, still photographs, or motion pictures). However, I would characterize this use of visuals as supplementing text or even replacing text altogether. The notion of visualization I'm pursuing here is more tightly coupled to emblematic representations of text itself. These graphics may be second- or third-order abstractions, synoptic views of knowledge rather than realistic representations of information (such as are film clips or photographs).

What I'd like to posit for this new literacy is somewhat akin to Ted Nelson's idea of "stretch text," in which the reader pans over planes of text, getting only an overview of the content. However, the reader has a level-of-detail joystick and can zoom in on anything that looks interesting. Traditional words and syntax do not lend themselves to the kind of squeezing and stretching Nelson imagined. (It would be difficult to imagine algorithms for this kind of real-time compression -- deleting all but the verbs and the nouns, for example, or perhaps a pseudo-parser that could distinguish between levels of specificity and generality in a paragraph might work.) It would be far easier to accomplish this compression in a visual mode. We already know, for example, that ideas can be compressed to objects, as is the case with icons. Certainly, the trend toward graphic user interfaces is a strong indication of the utility of visual information processing.

Let me digress for a moment here to talk about this trend. The bias in Western culture to measure information in pages and paragraphs leaves us with a generally naive understanding of graphics. Serious training in visual communication occurs only in such areas as graphic design, broadcasting, and various fields of entertainment. [41] [42] For other disciplines, theories of communication derived from the printed word predominate.

Nevertheless, a more formalized "rhetoric" of visual communication already exists in advanced science. While "graphic" equivalences for mathematical formulae have been standard ancillary forms of expression, some fields of science can only be comprehended in pictorial form: for example, the twisting, paired strands of DNA, brain maps, flight dynamics, and fluid-flow computations. [27] With the maturation of computer graphics, more exotic modes of cognition or new patterns for assessing phenomenological evidence may evolve. Surely, given the tremendous interest in scientific visualization and data-driven graphic, the notion of text-driven abstractions can't be far behind.

Graphical browsers are essential in overcoming the cognitive load of navigation through a body of text because the web becomes a higher-order abstraction of knowledge. And they have utility. A hypertext system allows for both hierarchical and non-hierarchical organizations of unstructured information. The same data may even exist in alternative structures, depending on the needs of the user. Also, global views (such as graphic browsers which visually present the web of links in any given hyperdocument) aid in the

top-down processing of information and increase comprehension. Such "meta-views" of data help the user to see logic patterns in a body of information and thus more easily assimilate meaning. This potential to make implicit knowledge structures explicit is one of the most exciting aspects of hypertext.

The hypertext graphical browsers I'm familiar with give only a crude picture of the logic structure of the web. Node content can be indicated by an icon or by some form of color coding. But it is not typical for the graphical browser to be fine-grained enough to report on quality or nature of the information in the node. Links can also be represented emblematically, giving type and perhaps attributes, but I don't know of any browser which can replicate the kind of skimming and scanning an expert reader uses to rapidly manage text. Nor do I know of any browser that can represent the visuo-tactile dimensions of prose style: characteristics captured in terms such as "texture," "voice," "density," "purple passage."

I'd like to see some enhanced forms of graphical browsers. For example, "pictorial abstraction" might become an innovative method for data compression once we get more experience with handling low-fidelity, non-representational knowledge constructs. Higher-order conceptualizations are not new to disciplines where the role of the computer has moved from number-cruncher to adjunct in conceptualization and creativity. Now imagine text representations (for say writings in history or philosophy) depicted through animation or pseudo-animation -- something on the order of abstract cartoons, fractal analysis imagery, or CAT scans.

Carrying the notions of information as environment and of knowledge as visual construct to their logical conclusion leads to virtual world technologies. Imagine, for example, the potential for a knowledge domain or "library land" (say late eighteenth-century European history) to be represented in three-dimensional space as stereoscopic features and figures. As a learner, you walk through the information environment -- or better yet, fly over the landscape -- looking for a particular subject. Since your assignment is to write a research paper on some aspect of the French Revolution, you land in Paris, zoom in on the knowledge shop in the Rue de Calais and walk in.

The friendly proprietor asks you a series of questions to determine your needs and then decides that, since you're just getting started on your project, it would be best for you to visit the Sculpture Gallery. Entering the majestic salon, you are amazed to see life-size figures of the major names associated with the period, clustered in order of importance. (Were you to find this grouping unsuitable for your purposes, you could select another arrangement -- alphabetical, let's say -- enter your request, and watch the clusters dynamically reconfigure.) As you stroll through the hall, you stop occasionally to activate a button at the base of each figure. The statue dissolves into a collection of icons representing the major subsets of information about this person. You place some of these icons on the automated notepad you are carrying.

After getting a feel for the period, you return to the shopkeeper and show her the items you have gathered. She helps you to refine and focus your collection of topics, and gives advice on gathering further information. Since you've now narrowed your focus sufficiently to make more detailed reading profitable, she directs you to the Hall of Records. This room reminds you of a well-stocked library, except that the items on the shelves are topic modules rather than titles of specific publications. The aisle on the French Revolution is organized alphabetically. However, as in the statue gallery, you can ask for a different configuration. You've decided to write on Danton and the Reign of Terror, and ask that only those items containing a set of keywords be visible. You've further requested that these items be rank-ordered based on a sophisticated algorithm that calculates the number of times the keywords appear and the size of the segment where the word appears.

Starting with the module that appears to be the best choice for finding quality information, you read an initial summary and begin to follow the links indicated by highlighted terms. Some nodes are visual -- a four minute video on Robespierre or a detailed genealogical chart on the French monarchy. An account of Danton's childhood is more comprehensive than you need, so you take the module in your hand and squeeze slightly. On reopening, you find that the text now presents information on a higher level of abstraction. Yet you still have the option of seeing the detail by selecting highlighted terms which then explode into a more fine-grained presentation. You've repositioned the content of your notepad onto a large, interactive workspace, and your icons now serve as a visual outline of the project. As you wind your way through the subject-matter modules, you can select segments and make links to your storyboard outline. You can also tag these links so as to indicate the nature of the relationship. This allows you to "view" your web from multiple perspectives. In other words, you might construct a web which links the materials chronologically, and you might -- using the same set of nodes -- construct a web that more clearly reflects an argumentative/analytical focus.

Since this is a labor-intensive project, you will spend several sessions at the terminal, each time returning to your workspace and further refining your paper as you continue to "experience" the research material. As you become comfortable with the subject -- and in fact become something of an expert in the content area -- you develop enough background to appreciate the major scholarly works. So at this point, you may want to read in their entirety relevant masterpieces such as Thomas Carlyle's multi-volumed history, The French Revolution, or Georg Buchner's play, Danton's Death.

Lest this start to sound like a science fiction version of Alice in Wonderland or a novel by William Gibson, let's briefly consider the state of some of the present-day technologies that will make the round book a reality for the twenty-first century. Dynamic, real-time, computer generation of complex imagery already exists. Well-financed disciplines, such as medicine and the military have had these capabilities for some time now -- as evidenced by various non-invasive diagnostics scanners and flight simulators or war game platforms. While the user can directly manipulate objects on the screen, the sensation is one of sitting at a window or a drawing board rather than being in the picture itself. Virtual world technology changes this by adding more realistic sensory cues. Stereo vision (binocular disparity) displays a different image perspective for each eye and produces 3-D representations. The Air Force uses two miniature CRTs mounted in a pilot's helmet to display a 3-D representation of the cockpit and external landscape. A device on the helmet detects head motion and dynamically updates the field of vision. Add a "data glove" or "data suit" -- devices which translate hand or body movements into mathematical coordinates that update a graphical screen in real time -- and the inhabitant of this artificial reality can directly manipulate objects in the visual field. Though these devices are still experimental, many practical applications are being planned. NASA, for instance, expects to use virtual worlds to expand the confines of the space station. Banks of electro-mechanical control panels can be computerized and literally done away with, yet their image can be represented as a virtual world.

Round Books in a WebWorld

It is unimportant whether hypertext makes possible the *integrated sensorium* of McLuhan's predictions or becomes the first step on the road to an electronically induced, hallucinogenic melding of mind and information structures -- the "cyberspace" so vividly depicted in William Gibson's science-fiction novels. The true benefit of hypertext will be as a universally available, interactive information environment in which users are taught to emulate the more powerful habits of mind characteristic of the experts.

Whereas square books are thoughts frozen in sequence, the round book becomes a rich environment to be traversed, manipulated, and experienced, much in the fashion of an adventure game. This is

more than just a reaffirmation of the benefits of "hands-on" modes of doing this. In the "integrated sensorium" of hypertext (and its extension, hypermedia), technological man can meaningfully enact the most pervasive, mythic scenarios of human existence. Learning becomes an extended journey, a passage from naivete to wisdom. Personified "experts" act as facilitators, gatekeepers, perhaps even grand inquisitors. Knowledge structures become virtual world landscapes -- perhaps fields of concepts, whose details can be harvested, winnowed, and stored. More primal, mytho-poetic intellectual constructs for ordering and explaining experience gain a new life. Allegories, quest motifs, riddles, adventure scenarios, and the like are resuscitated as vehicles for higher-order reasoning. Countless figures of speech used to discuss or describe learning -- the pale legacy left to typographic man of a time when knowledge came through a consortium of all the senses -- are now enacted -- much like an arcade game -- as serious parodies. Teaching strategies (e.g. the peripatetic classroom exercise) and heuristics (e.g. J. J. Gordon's various role-playing tactics for liberating the imagination presented in *Synectics*) -- things that we've traditionally used to compensate for the divorce from a plastic relationship with knowledge -- may no longer be necessary.

The world's current notion of literacy is based on the *square book*, a device which clearly has the potential to deskill its user. Additionally, much of what we now call "literacy" is proficiency in a cluster of skills necessary for success in state-supported education. Hypertext (and its extensions hypermedia) point toward a new metaphor: the *round book*, an n-dimensional, interactive environment, whose features reconfigure and grow to suit the needs and sophistication of the user. In essence, because of the symbiotic relationship between information and reader, the book becomes a cyberspace.

I'd like to thank Charles Hardwick, Glenn Freedman, and all the people associated with planning the conference for this opportunity to speak here today. I'd like to thank the audience for their indulgence -- particularly in some of the more fanciful speculations of my paper. More important, perhaps, I'd like to express my encouragement for all of us to continue to be pioneers in a technology that would appear to have a great deal to offer in these unsettled times.

Acknowledgements: *This paper was written during the author's tenure as a National Research Council Senior Associate at the Human Resources Laboratory, Brooks Air Force Base. The findings and opinions of the paper are those of the author and should not be construed as an official Department of the Air Force position or policy.*

Selected Bibliography

- [1] Boas, Franz. The Mind of Primitive Man. Westport, CT: Greenwood Publishing Company, 1983 reprint of the 1963 revised edition.
- [2] Bourne, John R. et al. "Intelligent Hypertutoring in Engineering," Academic Computing, September 1989, 18-20, 36-48.
- [3] Brand, Steward. The Media Lab: Inventing the Future at MIT. New York: Penguin Books, 1988.
- [4] Carlson, D. A. and Sudha Ram. "HyperIntelligence: The Next Frontier," Communications of the ACM, 33(3), 311-321.
- [5] Carlson, P. A. "Hypertext and Intelligent Interfaces for Text Retrieval," in E. Barrett (ed), The Society of Text: Hypertext, Hypermedia, and the Social Construction of Information, Cambridge, MA: MIT Press, 1989, 59-76.

- [6] Carlson, P. A. "Square Books and Round Books: Cognitive Implications of Hypertext," Academic Computing, April 1990, 16-19-26-31.
- [7] Carlson, P. A. "The Rhetoric of Hypertext," Hypermedia, 2 (2), Summer, 1990.
- [8] Colvin, Gregory. "The Current State of Text Retrieval," CD ROM: The New Papyrus. Microsoft Press, 1986, 131-136.
- [9] Dede, Christopher. "The Roles of Hypertext in Transforming Information into Knowledge," Proceedings of the National Educational Computing Conference, 1988, 95-102.
- [10] Dede, Christopher and Geetha Jayaram. "Designing a Training Tool for Imaging Mental Models," technical report for Air Force Human Resources Laboratory, Intelligent Systems Branch, February 1990.
- [11] Defanti, Thomas a. and Maxine D. Brown. "Insight through Images," Unix Review, 7(3), 42-50.
- [12] Diringier, David. The Book Before Printing. New York: Dover Publications, 1982.
- [13] Eisenstein, Elizabeth. The Printing Press as an Agent of Change: Communications and Cultural Transformation in Early-Modern Europe. Cambridge: Cambridge University Press, 1979.
- [14] Eisenstein, Elizabeth. The Printing Revolution in Early Modern Europe. Cambridge, England: Cambridge University Press, 1983.
- [15] Fairchild, Kim and Eric Gullichsen. "From Modern Alchemy to a New Renaissance." MCC Technical Report HI-400-86.
- [16] Fairchild, Kim and Alan Wexelblat. "Navigation through Cyberspace," MCC Technical Report Number STP-161-89.
- [17] Fox, Edward A. "Information retrieval: Research into New Capabilities," CD ROM: The New Papyrus. Microsoft Press, 1986, 143-174.
- [18] Furnas, George W. "Generalized Fisheye Views," Proceedings CHI'86, Boston, MA, April 13-17, 16-23.
- [19] Gill, Jock and Toby Woll, "Full Text Management," CD ROM: The New Papyrus. Microsoft press, 1986, 137-141.
- [20] Goody, Jack. The Domestication of the Savage Mind. Cambridge, England: Cambridge University Press, 1977.
- [21] Goody, Jack. The Logic of Writing and the Organization of Culture. Cambridge, England: Cambridge University Press, 1986.
- [22] Hartley, James. Designing Instructional Text Second. Ed. East Brunswick, NJ: Nichols Publishing, 1985.
- [23] Hashim, Safaa H. Exploring Hypertext Programming: Writing Knowledge Representation and Problem-Solving Programs. Blue Ridge Summit: Windcrest Books, 1990.
- [24] Hodges, Matthew, Russell M. Sasnett, and V. Judson Harward. "Musings on Multimedia," Unix Review, February, 1990, pp. 83-87.
- [25] Hull, Glynda. "Literacy as Prerequisite Knowledge," Proceedings of the 2nd Intelligent Tutoring Systems Research Forum, April 6-7, 1989.

- [26] James, Geoffrey. "Artificial Intelligence and Automated Publishing Systems," Text, ConText, and HyperText. Cambridge, MA: MIT Press, 1988, 15-24
- [27] McCormick, B. H., T. A. DeFanti, and M. D. Brown. "Visualization in Scientific Computing," Computer Graphics, 21 (6), 1-14.
- [28] McLuhan, Marshall. The Gutenberg Galaxy: The Making of Typographic Man. Toronto: University of Toronto Press, 1953.
- [29] McLuhan, Marshall. Understanding Media: The Extensions of Man. New York: McGraw-Hill, 1964.
- [30] Madigan, Chris. "Tools that Shape Us: Composing by Hand vs Composing by Machine," English Education, 16(3) 143-150.
- [31] Marshall, Catherine C. "Exploring Representation Problems Using Hypertext," Hypertext '87, pp. 253-268.
- [32] Nielsen, J. "The Art of Navigating through Hypertext," Communications of the ACM 33 (3), 298-310.
- [33] Olson, David R. "Computers as Tools of the Intellect," Educational Researcher (May 1985), 5-8.
- [34] Ong, Walter J. Orality and Literacy: The Technologizing of the Word. New York: Methuen and Co., 1982.
- [35] Searle, John R. "Is the Brain's Mind a Computer Program," Scientific American, January 1990, 25-37.
- [36] Solomon, Gavriel. "AI in Reverse: Computer Tools that Turn Cognitive," Journal of Educational Computing Research 4(2), 123-139.
- [37] Stedman, L.C. and C. F. Kaestle. "Literacy and Reading Performance in the United States, from 1880 to the Present," Reading Research Quarterly, 22, 8-46.
- [38] Swasy, Alecia and Carol Hymowitz. "The Workplace Revolution," Wall Street Journal Supplement: Education, Friday, February 9, 1990.
- [39] Talbert, Michael L. and David A. Umphress. "Object-Oriented Text Decomposition: A Methodology for Creating CAI Using Hypertext," Proceedings of the International Conference on Computer-Aided Instruction, 1989.
- [40] Thimmesch, Nick (ed). Aliteracy: People Who Can Read But Won't. Washington, DC: American Enterprise Institute for Public Policy Research, 1984.
- [41] Tufte, Edward. The Visual Display of Quantitative Information. Cheshire, CT: Graphics Press, 1983.
- [42] Tufte, Edward. Envisioning Information. Cheshire, CT: Graphics Press, 1989.
- [43] Vickery, B. C. "Knowledge Representation: A Brief Review," Journal of Documentation, 42(3), 145-159.
- [44] Zuboff, Shoshana. In the Age of the Smart Machine: The Future of Work and Power. New York: Basic Books, Inc., 1988.

N91-21956

A Knowledge Base Browser Using Hypermedia

Tony Pocklington

McDonnell Douglas Space Systems

Lui Wang

NASA/Johnson Space Center

Discussion of a hypermedia system we are developing to browse CLIPS knowledge bases. This system will be used to help train flight controllers for the Mission Control Center.

A Knowledge Base Browser Using Hypermedia

Tony Pocklington
MDSSC-Houston/MDCB2KI

Lui Wang
NASA-JSC/PT4

Abstract

Currently under development at the Johnson Space Center (JSC) for use in the Mission Control Center (MCC) are a group of expert system to assist the ONAV flight controllers. These expert systems serve two functions: to act as an assistant during real-time and simulated flight operations, and to act as trainers for controllers when flight operations are not in progress. The Knowledge Base Browser (KBB) is a tool also currently under development. The goal is to augment the expert system in its role as a trainer.

The KBB will take advantage of the structure of the rule base. The rules and relation patterns are the basic nodes and links of the hypermedia system. Meta links and collection nodes can also be used to further organize information for the loosely structured rule base. Browsing this knowledge base will be accomplished either by navigating through the various collection nodes that have already been defined, or through a query language.

1.0 Introduction

Currently under development in the Mission Control Center (MCC) at the Johnson Space Center (JSC) is a collection of expert systems to support the Onboard Navigation (ONAV) flight control position. The primary function of these expert systems is to act as an assistant to the ONAV flight controllers, who support the Ascent, Rendezvous and Deorbit/Landing Phases of a Shuttle mission.

Along with the expert systems are several other programs under development, including a data logger, a playback program, a program to display analog data, a plot program, and the Knowledge Base Browser (KBB) that is the focus of this paper.

2.0 Reasons For Browser

The KBB will serve two purposes: to assist in the verification of the rule bases for the various expert systems, and to augment the training of the flight controllers.

2.1 Verification

The verification of an expert system is a difficult task. In a rule base language, each rule is an independent entity. As the number of rules increases, the interaction of the rule base is more difficult to track and hence more difficult to verify. A typical knowledge base application, such as the Entry ONAV Expert System which has in excess of three hundred rules, needs a browsing mechanism that could simplify the process of inspecting these rules. One of the functions of the KBB will allow users to examine the causal relationship of the rules and facts in a rule base language.

2.2 Training

The KBB will also be an excellent training tool. A great deal of work goes into the development of a rule base, and implementing browse and query functions will allow these rule bases to be treated as experts. Novice flight controllers will be able to use the browser to read through all or

part of the rule base, while more experienced users will probably find the query capability useful to answer specific questions. For example, a relatively new controller might want to read all rules related to inertial measurement unit (IMU) failures, whereas an experienced user would be more interested in a rule regarding the reselection of an IMU which had had a prior communications fault.

3.0 CLIPS

The ONAV expert systems were developed using the 'C' Language Integrated Production System (CLIPS). CLIPS is a forward chaining rule base production system developed by the Software Technology Branch at NASA/Johnson Space Center. CLIPS has capabilities similar to OPS5 (Official Production System), and ART (Automated Reasoning Tool). The purpose of the tool was to address delivery of expert systems to conventional operational environment.

3.1 CLIPS Knowledge Base

In a knowledge base system like CLIPS, the notion of nodes and links are implicitly embedded in the rules. The fact patterns are viewed as links, and the rules are viewed as nodes. Each rule contains partial knowledge of the overall system and acts opportunistically based on the incoming data stream. Rules in general define how to transition between the different states. The antecedents of the rules capture the current events, and the consequents of the rules modify the system and take it to a new state. The rule nodes are one of the atomic units in the KBB. This is because each rule is syntactically an independent unit, but it only conveys partial information. The fact patterns, on the other hand, are the agents that link all the rules together. Therefore the fact patterns are the atomic links for the KBB.

3.2 CRSV

Some of the data structures for the KBB were taken from the Cross-Reference, Style and Verification (CRSV) utility. Some of the ideas for the verification part of the KBB came from CRSV as well.

CRSV is a tool that was also developed at JSC to help verify the CLIPS rule bases. The focus of this tool is to address the software engineering practice in rule base programming. For example, the tool detects and issues warning and error messages for "bad" programming style, syntax errors, and inconsistent data type. It also performs cross referencing among relations and variables. In addition, CRSV collects statistical information that may help developers to improve the system performance. Even though the target users and the purpose of the two tools, CRSV and KBB, are different, they share one major common function, which is the cross referencing or the browsing function. This function carries different meaning based on the users' perspective. The rule base programmers use the cross referencing function to verify facts assertions and retractions to the fact base. The domain experts use the browsing function to verify the completeness of the specification. Finally, the novice trainers use the browsing function to understand the causal effects of the system behavior.

4.0 Browsing

One of the most important features of the KBB will be the browsing function. The two means of browsing the knowledge base will be navigating through the various collections of rules and making queries of the rule base.

4.1 Navigation

The user will be given a list of collections, which may contain either rules or other collections of rules. Selecting a collection will cause a new list to be displayed, which will either be rule names or the names of more collections. A window displaying where the user is in the tree hierarchy of collections will also be available to keep users from getting lost.

4.2 Queries

The other means of traversing the rule base will be a query function.

4.2.1 Query Language

There will be a very basic query language available for the users. The query language will allow users to do simple searches on text strings and to combine these queries using logical operators.

4.2.2 Intelligence

It may be necessary to build some knowledge into the query language parser, in order to handle certain context problems. For example, suppose a user wants to view all rules pertaining to TACAN data being inhibited. A natural search string that he would then formulate would be "tacan inhibit". The actual string that must be searched for, however, might be "tacan-aif-pass inhibit". Since the word "inhibit" might also appear as the value of a variable, it might be necessary to search first on the words "tacan-aif-pass", and then apply rules to eliminate instances that did not either have the word "inhibit" or a variable whose value might be "inhibit". The rules to handle these searches will probably take the form of another CLIPS knowledge base, allowing users to quickly alter the rules without having to recompile and relink the system.

4.2.3 Nested Queries

The system must also have some means of handling queries that return either too much or too little data. One way to handle this is to allow nested queries, where the data returned from one query is used as the search space for the next query. It must also be possible for users to back up, either partially or all the way to the beginning level, where the entire rule base is used as the search space.

5.0 Scripting

The final part of the system is the scripting, or creation of the collections. The easiest way to create a collection is to save the results of a query. The user will be given the opportunity to supply a name for the new collection and save it to the system, either as one of the main collections at the root level of the system, or as a subcollection of one of the larger collections.

Also, it is possible that the results of a query the user wishes to save are too large for a single collection. The user will have the capability in such instances to select elements individually to be saved into subcollections.

6.0 Conclusion

When it is complete, the KBB will be a versatile tool for the verification and browsing of CLIPS rule bases. Some of the features of this hypermedia system will be the automatic creation of links based on the CLIPS rule structure, the ability to query the rules and save the result as a collection, and the ability to browse the rules, either sequentially or by using the links and collections.



Hypertext As a Model for the Representation of Computer Languages

Randal Davis
University of Colorado

Computer systems for operating the Space Station Freedom will include an object-oriented and English-like User Interface Language (UIL). We have proposed a representation of the Space Station UIL that is based on a hypertext model. We discuss the hypertext model of the Space Station UIL and show how this representation may be appropriate for other modern computer languages.

Hypertext As a Model for the Representation of Computer Programs

Experience with the Design of the
Space Station Freedom User Interface Language

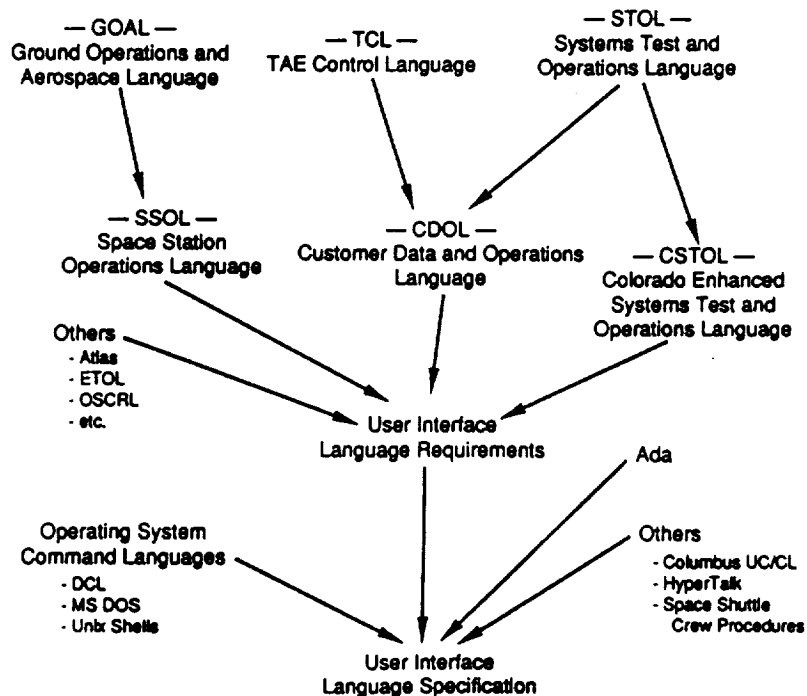
Randal L. Davis
Space Operations and Information Systems Division
Laboratory for Atmospheric and Space Physics
University of Colorado at Boulder

Presented at
Hypermedia '90
Aerospace Applications and Research Directions
Houston, Texas
5 December 1990

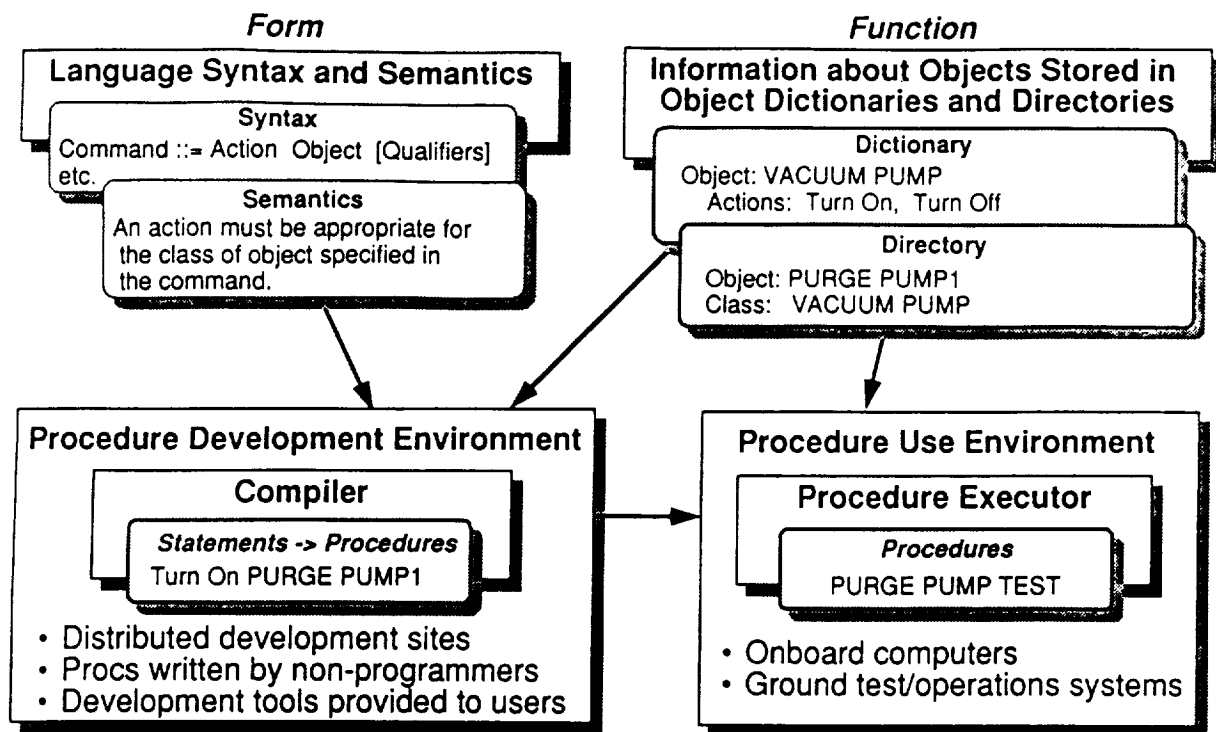
Introduction

- The Space Station Freedom User Interface Language (SSF UIL) is designed for use by the astronauts, ground controllers, scientific investigators, and hardware/software engineers who will test and operate the systems and payloads aboard the space station
 - Object-oriented
 - English-like
 - Will supplement the graphical user interface to systems and payloads by providing command line entry
 - Will be used to write test and operations procedures
- The SSF UIL design was influenced by the availability of new technologies, including hypertext
- We have found at least three places where hypertext is appropriate for use within the systems that will be used to create and run SSF UIL procedures

The SSF UIL Family Tree



Overview of SSF UIL Design and Usage



Hypertext Application #1: Linking Code and Annotation

- Provide a link between units of code — statements, steps and procedures, and so on — and associated annotation and documentation
 - History of procedure's development [Development, Use]
 - Description of syntax and semantics [Development, Interactive Use]
 - Comments for programmers [Development]
 - Help for end users [Use]
- Why Hypertext?
 - Freeform comments aren't good for capturing specific information like program history
 - Intertwining code and commentary often makes a procedure more difficult for a skillful procedure developer to read
 - Comments don't survive parsing, so they aren't available to users who only have the object or executable representation of a procedure
 - Comments are formed from the character set used for the computer language, but it would often be desirable to allow graphics and other non-text information in comments

Sample Procedure With Traditional Freeform Commentary

.....
Procedure PURGE PUMP TEST - This procedure performs a full checkout of the ECLSS purge pumps. Run immediately after maintenance to the purge system.
.....

Code Documentation
and User Instruction

Written by Kevin Smith, 1995/5/20
Modified by Jennifer Thomas, 1996/11/14: Updated to handle the new ACME 301 J-series pumps.
.....

History

Procedure PURGE PUMP TEST Is

Declare PUMP: VACUUM PUMP /* Current pump under test */
Declare DESIRED SPEED: ANGULAR VELOCITY:=1000 RPM

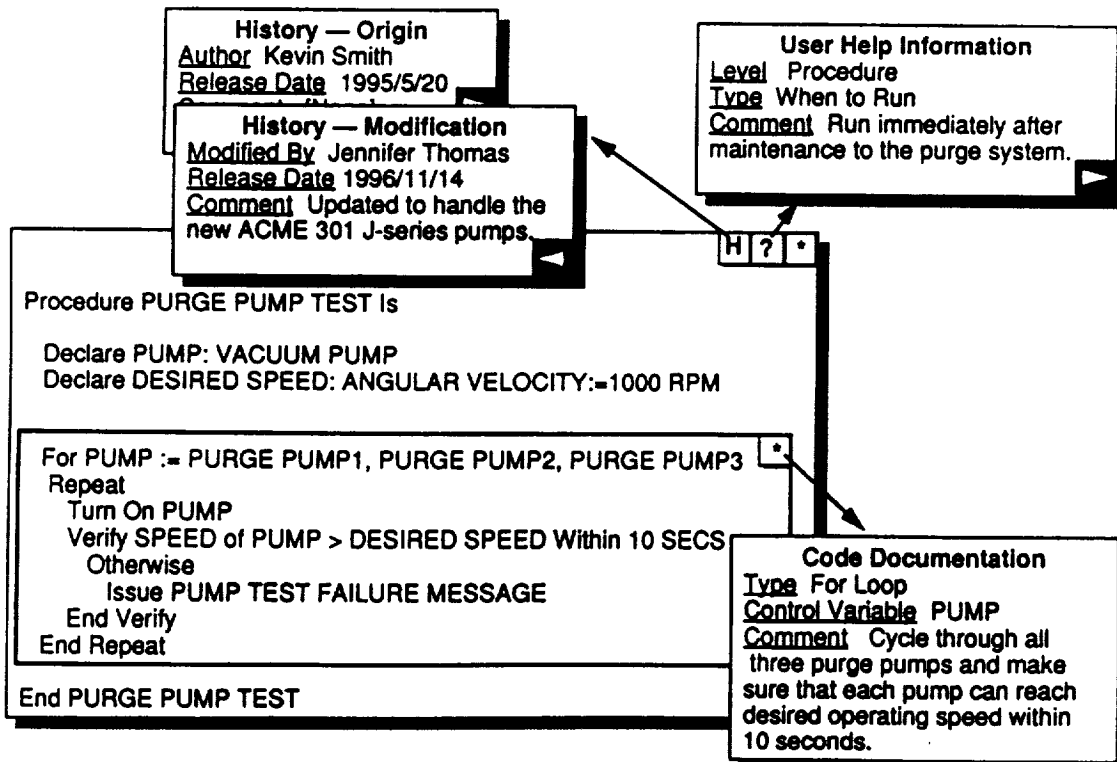
Code
Documentation

/* Cycle through all three purge pumps and make sure that each pump can reach desired operating speed within 10 seconds */

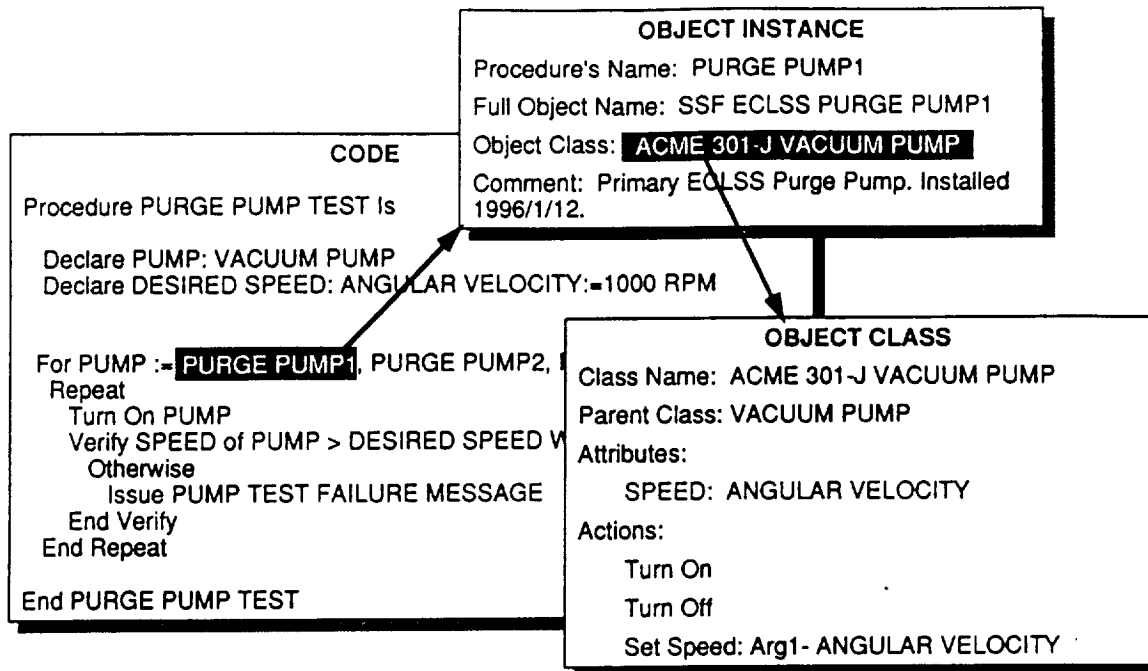
For PUMP := PURGE PUMP1, PURGE PUMP2, PURGE PUMP3
Repeat
Turn On PUMP
Verify SPEED of PUMP > DESIRED SPEED Within 10 SECS
Otherwise
Issue PUMP TEST FAILURE MESSAGE
End Verify
End Repeat

End PURGE PUMP TEST

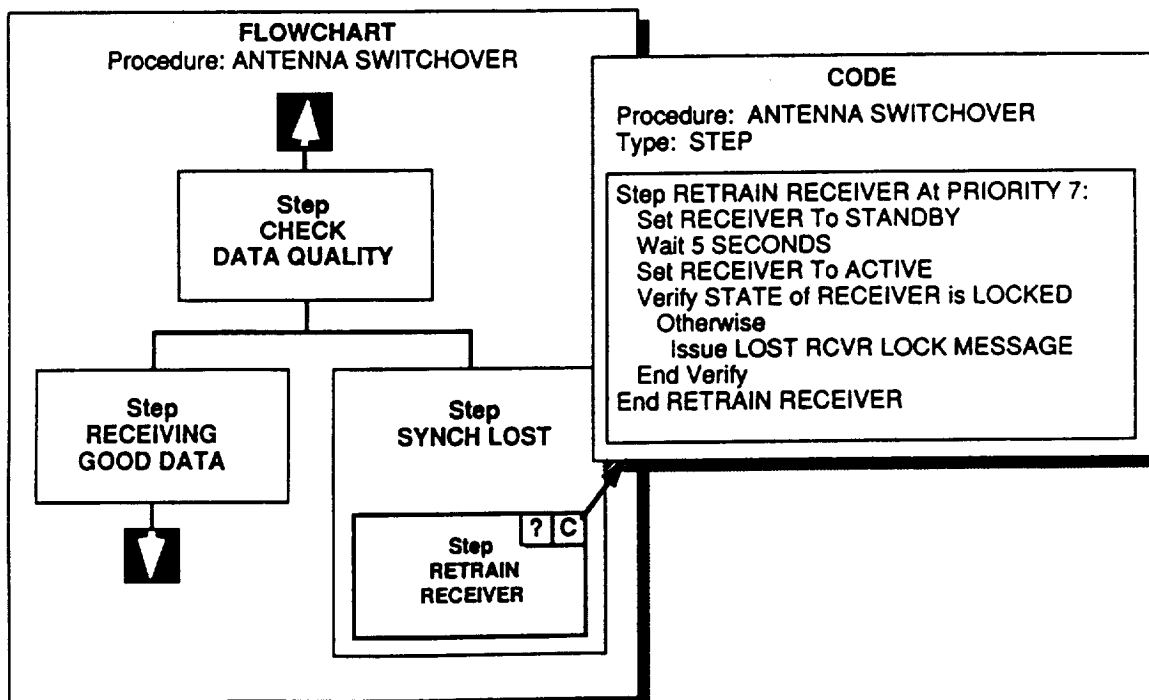
Procedure With Hypertext Annotation



Hypertext Application #2: Linking Code to Object Information



Hypertext Application #3: Linking Steps Within a Procedure



Conclusions

- Hypertext is appropriate and advantageous for the three uses we have examined:
 - Linking procedure code and annotation
 - Linking code to object instance and class information
 - Linking steps within a procedure
- The SSF UIL's object-oriented nature lends itself to representation through hypertext
- While the SSF UIL was specified from the outset with hypertext in mind, it has become clear that a traditional text-only representation for procedure code and annotation is desirable to promote portability
 - An annex is being added to the SSF UIL Specification to provide this
- Hypertext-based organization will be appropriate for other modern languages, particularly if they are designed from the outset to take advantage of new technology like workstations and personal computers

References

- **SSF UIL Documentation**
 - Space Station Freedom Program: *"User Interface Language Specification"*, Document No. USE 1001, Version 2.1, March 1990.
- **Use of Hypertext for Program Documentation**
 - Bigelow, J.: "Hypertext and Case", *IEEE Software*, March 1988, pp 23-27.
 - Wolfram, S.: *"Mathematica — A System for Doing Mathematics By Computer"*, Addison Wesley, 1988.
- **Concepts for Crew Procedures, Old and New**
 - Johns, G. L.: *"Flight Data File for the Space Station"*, 2 Volumes, MITRE Corporation, Document No. MTR10019, February 1987.
 - Johns, G. L.: *"Dynamic Display of Crew Procedures for Space Station"*, 2 Volumes, MITRE Corporation, Document No. MTR 88D0033, August 1988.
 - Johnson Space Center: *"Space Shuttle Flight Data File Preparation Standards"*, Document No. JSC-09958, May 1984.
 - Kelly, C. M.: *"Conceptual Definition for a Flight Data File Automated Control and Tracking System"*, MITRE Corporation, Document No. MTR-88D0017, July 1988.

Acknowledgement

This work was supported by NASA Goddard Space Flight Center under contract NAS5-29174. Design of the SSF User Interface Language was coordinated by the Space Station Freedom User Support Environment Working Group (USEWG), and many USEWG members have contributed to the language's development.



Session 6

Hypertext and Object Management


Chair: Bryan Fugate

AI GERM: A Logic Programming Front End for GERM

Safaa H. Hashim

HEAVENS System for Software Artifacts

Paul Matthews



V91-21957

**AI GERM: A Logic Programming Front End
for GERM**

Safaa H. Hashim

AiGerm: A Logic Programming Front End for Germ

Safaa H. Hashim

MCC/STP

**Microelectronics and Computer Technology Corporation
Software Technology Program
3500 West balcones Center Drive,
Austin, TX 78759-6509
INTERNET: hashim@mcc.com**

Introduction

AiGerm (Artificially Intelligent Graphical Entity Relation Modeler) is a relational database query and programming language front end for MCC/STP's Germ (Graphical Entity Relational Modeling) system. Currently, three versions of AiGerm are in use: Quintus Prolog, BIMprolog, and LDL (MCC's Logical Data Language). AiGerm is intended as an add-on component of the Germ system to be used for navigating very large networks of information, harnessing Prolog or LDL's relational database query capabilities. It can also function as an expert system shell for prototyping knowledge-based systems. AiGerm provides an interface between the programming language and Germ.

When a user starts up AiGerm, the system builds a knowledge base of the currently loaded Germ folio. The knowledge base is a collection of node, link, and aggregate facts. Selecting from the set of commands built in to the AiGerm interface, the user can query the database and run programs that select, create, delete, inspect, and aggregate the nodes and links appearing in the Germ browser.

Aigerm is currently used in MCC/STP's DESIRE system to extract information on the design of code for software systems. Members of the research staff are experimenting with AiGerm in building IBIS-based reasoning and decision support systems for software design and engineering. Rockwell International, an MCC/STP shareholder, is using AiGerm in a simultaneous engineering project.

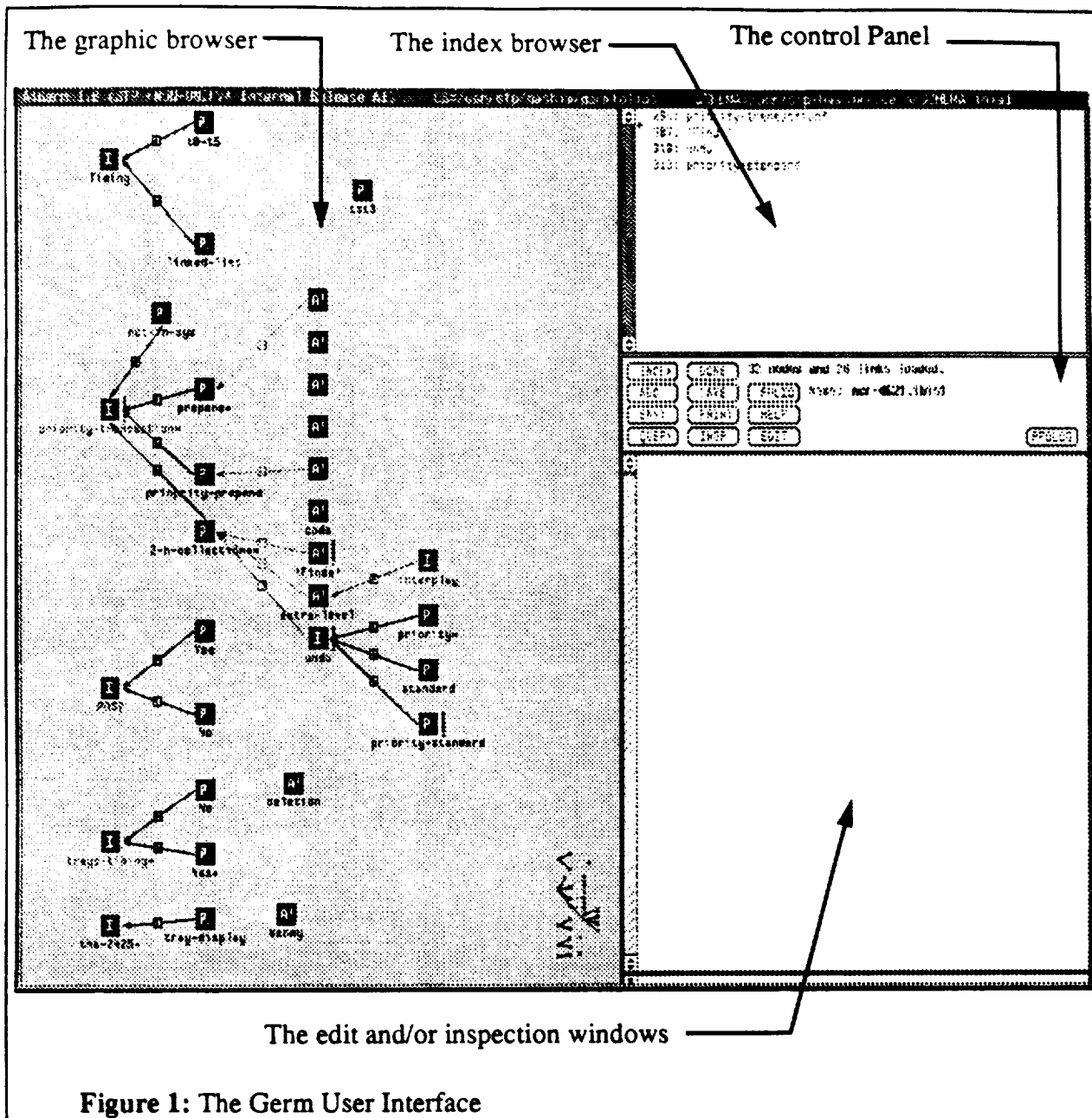
What is Germ?

Germ (Graphical Entity Relational Modeler) is a graphically-oriented tool for browsing and editing databases. What distinguishes Germ is its conceptual approach in abstracting the elements of a database. Germ uses a few abstractions that we can easily comprehend, remember, and use to create, understand, retrieve, and manipulate database objects. There are two sets of such concepts: basic concepts (also known as object concepts) and interface concepts.

Germ applications are based on an underlying schema file that defines Germ objects and their behavior. The basic object types of the Germ schema are: nodes, links, collections, and aggregates. An application based on a given Germ schema is called a folio; many folios can be based on the same schema. The schema contains the declarations for most of the object concepts in Germ. Embedded in the schema object concepts are properties such as shape, color, attribute types, and so on. Together, the object concepts in a given schema file represent a method for modelling a certain problem, understanding it, and solving it.

Germ's "interface concepts" include a set of window objects: a graphical browser, global view, index window, control panel, inspection window, and editing window, see Figure 1. These objects allow the user to interact with the system to add, delete, update, and retrieve information

represented graphically as nodes and links and to query the database.



Germ Applications

In its current form, Germ is a generalization of gIBIS. In gIBIS (Conklin & Begeman, 1988), the network of entities represents the argumentation process for understanding and solving a problem using the IBIS methodology. IBIS (Issue Based Information Systems) was introduced by Horst Rittel in the late sixties and early seventies (Rittel & Webber, 1969; Kunz & Rittel, 1970). We have reimplemented gIBIS in Germ just by using a special schema file representation of the IBIS method. The graphic browser in Figure 1 shows a gIBIS network implemented

using a Germ schema file for gIBIS.

Using Germ, researchers and system designers (working individually or in groups) can model systems derived from any method, not just IBIS. In the case of gIBIS, we can have different versions of the gIBIS system that are based on different versions of the gIBIS schema file, each version representing variations in implementing the IBIS method.

Germ can represent both a given model of a method and the database of information on which the method is based. The method can be a design, a problem understanding method, or a problem solving method. Germ will probably be used mostly for building a database representing a problem solving method (problem solving presupposes problem-understanding, and design methods are a special subset of problem solving methods).

Germ is so generalized that it could be considered as a graphical tool that uses geometrical shapes and text to present documents and designs. This is why Some STPer's think of Germ as a "GEometrical Relation Modeler". Germ contains a set of on-line tutorials on Germ usage that were developed using Germ itself. In this case, Germ was used as hypertext-document writing tool (Garrison, Marks and Creemer, 1989). In this article we consider Germ as a modeling tool.

Why AiGerm

Germ has its own query mechanism, which is inflexible for a number of reasons, the two most important ones being:

- Its keyword search combined with regular expression pattern string matching allows only simple queries, like those shown in the preceding section. More importantly, the expressive power of these simple queries is very limited, such that the following simple query is not possible:

Find the issue with the word "interface" as part of its contents and at least one position responding to it.

In Prolog, on the other hand, this query would be easily expressed in a single query(goal):

```
|?- issue(I),contents(I,C),substring_of("interface",C),
responds_to(P,I).
```

Of course, for a practical and real design or an engineering application we would need more complex queries. This requirement, which can be easily met using Prolog, is known as the problem of "structure search" in hypertext (Halasz and Conklin, 1989).

- In addition to richer expressive power in a query mechanism, we need an inference engine, which is a must in the design and engineering tasks of today. The current version of Germ provides no inference engine. With even such a simple one as that in Prolog, we can transform Germ into a powerful knowledge engineering system.

AiGerm is designed to address these two deficiencies in the current query mechanism in Germ. This is why we currently define AiGerm in the following way:

AiGerm = Germ + Logic Programming

A Review of AiGerm

To use AiGerm, the user must have Germ running on a local or a remote machine. Before starting AiGerm, the user must start up Germ and load the desired (hypertext network) folio into the Germ browser. Then, in a shell window, the user would give the command:

```
AiGerm <HOSTNAME>
```

where HOSTNAME is the name of a remote workstation. If no HOSTNAME is given, AiGerm interfaces to the Germ system running on the local workstation. Before actually starting the Prolog process, AiGerm builds a Prolog knowledge base file, see Figure 2.

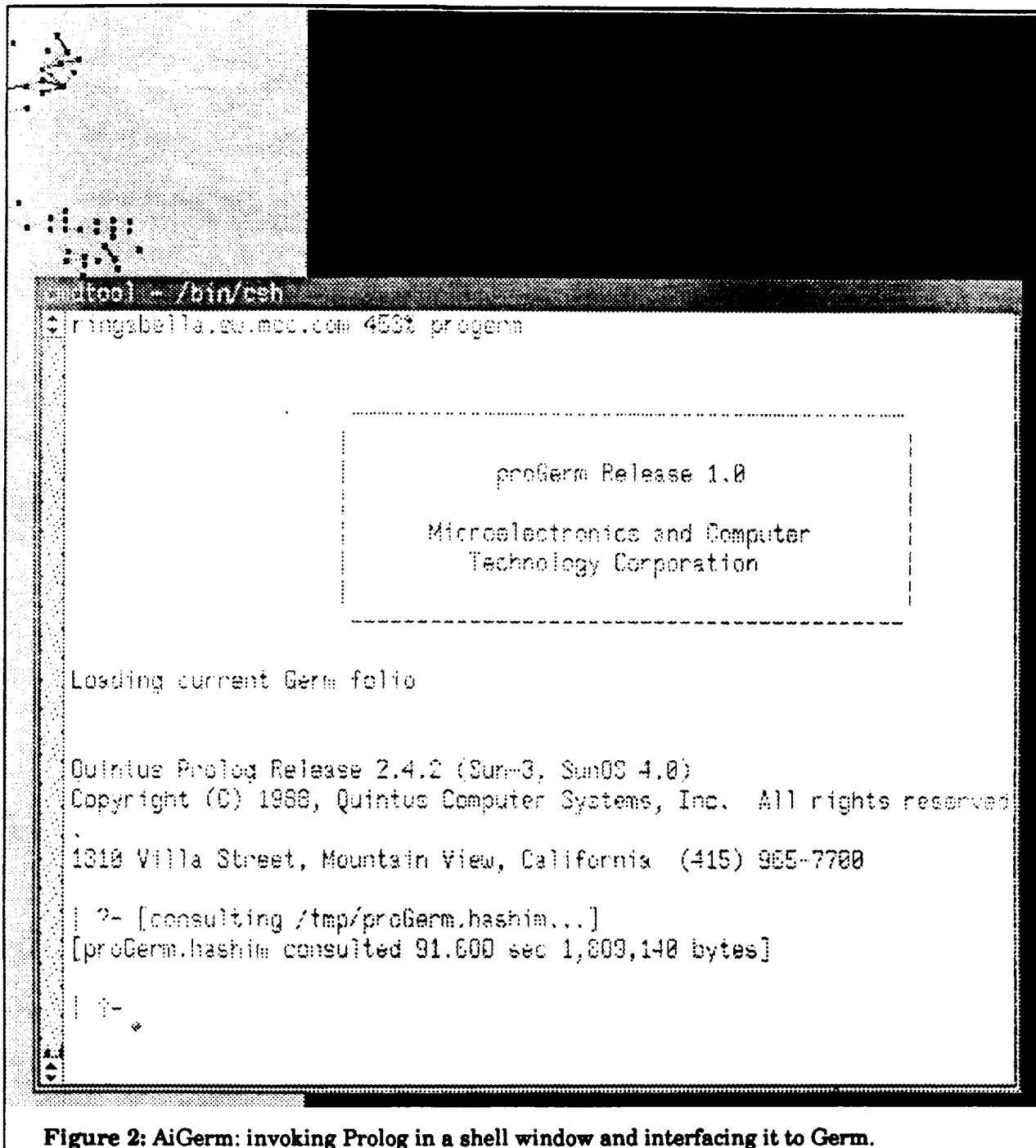


Figure 2: AiGerm: invoking Prolog in a shell window and interfacing it to Germ.

In this knowledge base, for each hypertext entity—i.e. node, link, and aggregate—AiGerm asserts a fact (a prolog clause). Once the knowledge base file is complete, the Prolog process is started and is directed to consult the knowledge base file. When this knowledge base is loaded

into Prolog, nodes, links, and aggregates are represented as Prolog facts, also known as base-relations. The abstract forms of these facts are:

```
node(Eid,[ATTR, ATTR, ...]).
Link(Eid,[EID,EID],[ATTR, ATTR, ...]).
agg(Eid,[EID,EID, ...],[ATTR, ATTR, ...]).
```

As

EID = the compound term "eid(INTEGER)"

ATTR = the compound term "attr(TYPE,VALUE)"

where TYPE and VALUE are:

TYPE = label;author;date;sid,subject;keywords;and so on,

VALUE = STRING;INTEGER

Following are examples of a node, a link, and an aggregate, each represented as a fact:

```
node(eid(293),[attr(type,"issue"),
               attr(sid,1),
               attr(date,"Jun 8 10:14 1989"),
               attr(author,"Kemp"),
               attr(label,"Timing"),
               attr(resolved,"yes"),
               attr(contents,"How are timings from multiple trays handled?"),
               attr(x,70), attr(y,36)]).
```

```
link(eid(314),[eid(294),eid(293)],[attr(type,"responds-to"),
                                   attr(sid,-1),
                                   attr(date,"Jun 8 10:17 1989"),
                                   attr(author,"Klempay")]).
```

```
aggr(eid(293),[eid(293),eid(295),eid(294)],[attr(type,"AGI")]).
```

Using Prolog to Query Germ Networks

We can query a Germ network directly by issuing goals at the top-level system prompt (|?-). For example, to retrieve nodes one at a time we give the goal:

```
| ?- node(X,List).
```

and Prolog will return the first instance of node that matches this goal, namely:

```
X = eid(7),
List = [attr(type,"issue"), attr(sid,43), attr(date,"May 26 18:23 1989"),
        attr(author,"hashim"), attr(label,"theory"), attr('Resolution-due-
date',"Jan 1 1990"), attr('Contents',"^J^JWhat kind of IBIS-theory are we
after?^J")]
```

Retrieving node and link facts is useful but not very interesting. The advantage of Prolog queries over the standard (static) Germ query system becomes apparent when we start giving Prolog sequences of connected subgoals. For example, we can use Germ to model the IBIS method in a way similar to that of the gIBIS system. We would then have a structured hypertext network of issues, positions, and arguments for capturing, say, a group problem-solving or a design meeting session. For real world applications, an IBIS network could have hundreds of nodes and links representing the different issues, positions, and arguments and their relationships. Navigating such large networks is quite difficult if it is done manually. On the other hand, in AiGerm we can use Prolog to query the network for certain nodes and links. For example, we can give this query:

```
| ?- node(X,List), member(attr(type,"issue"),List).
```

meaning that we want to retrieve only nodes that are issues. Moreover, we want to highlight the issue nodes on the browser canvas while retrieving them. To do that we can write this compound goal:

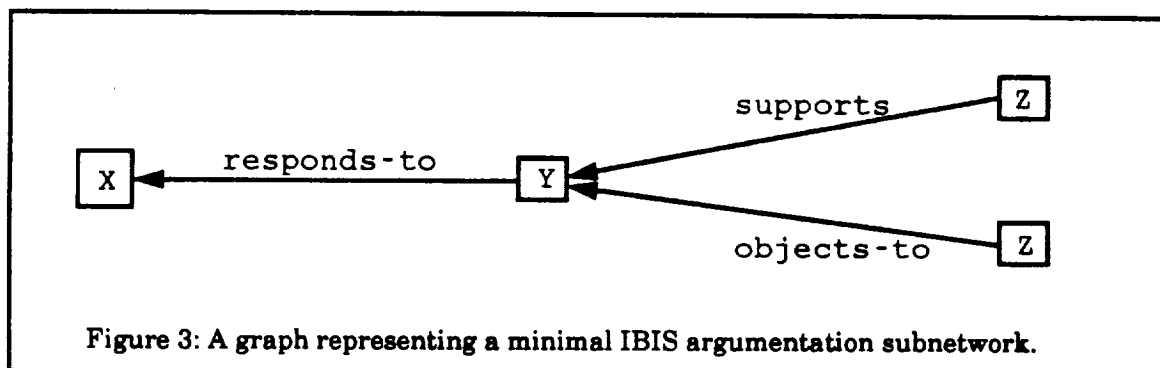
```
| ?- node(X,List), member(attr(type,issue),List), hl_eid(X).
```

hl_eid is an add-on (built-in) predicate for interfacing Prolog to Germ. A more interesting goal is to retrieve a more structured set of nodes; for example, to verify that our design discussion satisfies this minimal argumentation subnetwork condition: our IBIS network must have at least one issue with at least one position responding to it, and there must be at least two arguments, one supporting the position and the other objecting to it.

A graph representation of such a subnetwork is shown in Figure 3. Here is the Prolog query for such a structure:

```
| ?- node(X,XNodeAttList),
    member(attr(type,"issue"),XNodeAttList),
    link(L1,[Y,X],LinkAttList1),
    member(attr(type,"responds-to"),LinkAttList1),
    node(Y,YNodeAttList),
    member(attr(type,"position"),YNodeAttList),
    link(L2,[Z,Y],LinkAttList2),
    ( member(attr(type,"supports"),LinkAttList2);
      member(attr(type,"objects-to"),LinkAttList2) ),
    node(Z,ZNodeAttList),
    member(attr(type,"argument"),ZNodeAttList),
    hl_eids([X,L1,Y,L2,Z]).
```

The last subgoal, namely the predicate hl_eids, takes a list of entity EIDs and highlights (selects) them. Suppose we have a compound goal—that is, a goal made of a sequence of subgoals—that we might need to fire later or use as a subgoal in yet another compound goal. It is worthwhile in such a case to capture a query into a rule (a program) that stands for an executable definition of an “abstraction.” This brings us to the subject of abstracting new concepts from existing ones in hypertext networks.



Deriving New Abstraction from Existing Germ Networks and Other Abstractions

Enhancing Germ’s hypermedia query and navigation capabilities is not the only advantage of using the logic programming interface in AiGerm. Another advantage is the ability to define new abstractions from the existing pool of base relations and other previously defined abstractions. We say “new abstractions” because Germ itself, through our schema file definition, allows us to have an initial (built-in) set of abstractions on top of the basic node, link, collection,

and aggregate primitives. For example, using a schema file to represent the IBIS method, we usually have abstractions for issues, positions, arguments, and their relationships defined in terms of nodes and links. The knowledge base that AiGerm builds for a Germ network is basically made up of node, link, and aggregate facts. From these facts we can easily define the first level of abstractions as follows:

```
/* ***** issue ***** */
/* flow-pattern: (i), (o) */
issue(EID):-
    node(EID,ATTlist),
    member(attr(type,issue),ATTlist).

/* ***** position ***** */
/* flow-pattern: (i), (o) */
position(EID):-
    node(EID,ATTlist),
    member(attr(type,position),ATTlist).

/* ***** argument ***** */
/* flow-pattern: (i) (o) */
argument(EID):-
    node(EID,ATTlist),
    member(attr(type,position),ATTlist).
```

For the relationships (links) between issue, positions, and arguments, we can define the *responds-to*, *supports*, and *objects-to* relationships in a similar way. For example, here is a definition of the active relationship *responds-to* between a position and an issue that is supported by Germ:

```
/* ***** responds_to ***** */
responds_to(P,I):-
    link(_, [P,I],ATTlist),
    member(attr(type, responds-to),ATTlist).
```

What is not supported by Germ is a passive version of *responds_to*, which we can easily define in Prolog as *responded-to-by*:

```
/* ***** responded_to_by ***** */
responded_to_by(I,P):-
    responds_to(P,I).
```

Similarly, we can define *objects-to*, *objected-to-by*, *supports*, and *supported-by* link types. In essence, we can explicate the methodology implicit in a Germ schema file by using such rules. Moreover, we can extend the schema definition in a more flexible way than directly editing and changing the schema file itself. Thus, we can define special modified views of the schema (and thus the methodology represented by the schema) without imposing on other people using the same schema. This ability to modify the representation in such an interactive and dynamic way is a basic aspect of AiGerm.

The abstractions discussed here are just one level above the entity-relation model representation. We can have abstractions that are made up of other abstractions, which themselves are made up of other abstractions, and so on. An example of a system-model using such a multi-level abstracting technique is the following representation of an IBIS-network:

```
% Each IBIS issue must have at least two lines of arg. SL and OL
ibis(I, [SL,OL|REST]):-
    issue(I),
    sup_argLINE(I,SL),      % supporting line of argumentation
    obj_argLINE(I,OL),      % objecting line of argumentation
    ibisl(I,REST).
```

```

ibis1(I, [LINE|REST]):-          % it can have other argumentation lines
    argLINE(I, LINE),
    ibis1(I, REST).
ibis1(_, []).

```

This definition of an IBIS subnetwork requires that an issue have at least two lines of argumentation, a supporting line and an objecting line. But supporting and objecting lines of argumentation are just special kinds of the argLINE abstraction:

```

argLINE(I, LINE):-
    sup_argLINE(I, LINE).      % a supporting line of argumentation
argLINE(I, LINE):-
    obj_argLINE(I, LINE).      % an objecting line of argumentation
argLINE(I, LINE):-
    cha_argLINE(I, LINE).      % a challenging line of argumentation

```

For the three special lines of argumentation we can have the following definitions:

```

sup_argLINE(I, [P,A|REST]):-
    issue(I), position(P, I), responds_to(P, I), supports(A, P),
    argSEQUENCE([A|REST]).
obj_argLINE(I, [P,A|REST]):-
    issue(I), position(P, I), responds_to(P, I), objects_to(A, P),
    argSEQUENCE([A|REST]).
cha_argLINE(I, [I1|REST]):-
    issue(I), issue(I1), suggested_by(I1, I),
    argSEQUENCE([I1|REST]).

```

To complete our sequence of abstractions, we need to define argSEQUENCE, which stands for a sequence of argumentation moves:

```

argSEQUENCE([A,A1|REST]):-
    supports(A1, A), argSEQUENCE([A1|REST]).
argSEQUENCE([A,A1|REST]):-
    objects_to(A1, A), argSEQUENCE([A1|REST]).
argSEQUENCE([A,I|REST]):-
    suggested_by(I, A), argLINE(I, REST).
argSEQUENCE([_]).

```

We believe that such high-level abstractions make navigating Germ networks much easier than navigation with just the basic nodes and links. Also, it makes more sense to talk about related abstractions, such as "a position responding to an issue," than just talking about independent unit abstractions, such as issues, positions, and arguments. For example, issues, positions, and arguments are elements of a discussion or a discourse. Related abstractions form representation structures which we could use to express complex theories and methods. The "ibis" predicate is such a structure that we can use to model the IBIS-based system design process. As a result, we expect that prototypes of system (both software and hardware) engineering applications can be built more efficiently and rapidly using AiGerm's combination of visual modeling in Germ and abstraction-based representation in logic programming. The next section reports on a number of AiGerm-based applications in software engineering and engineering system design.

AiGerm Applications

While we are still in the early stages of experimenting with AiGerm, we feel that it in addition to its use as a relational database query-based hypermedia system, AiGerm could be equally viewed as a general and cost effective tool for prototyping AI-based hypermedia systems. It is this prototyping ability of AiGerm for which we anticipate multiple applications. Currently we are exploring:

1. Reasoning with Issue-Based Design Rationale Networks
2. Analyzing the Structure of Programs
3. The Intelligent Documentation Experiment

Also, researchers at the **Space Systems Division** of **Rockwell International** are currently using AiGerm in developing research prototypes for:

1. QFD Expert System Research
2. Simultaneous Engineering Environment
3. Design Reuse project
4. Design Decision Support prototypes
5. Knowledge Capture
6. Requirement's Analysis (NASP)
7. Payload Mfg. Cost Analysis and Design
8. CAD/CAM Expert system Technology

To illustrate how AiGerm can be used for prototype development, we present two examples in the sections that follow: the "reasoning with IBIS" example and Rockwell's "QFD expert system" example."

EXAMPLE 1: Reasoning with Issue-Based Design Rationale Networks

Although logic programming is based on formal logic, we believe it can also be used for exploring other modes of reasoning, both formal and informal. We have identified four non-mutually-exclusive reasoning methods that we can apply to the IBIS method:

1. a formal reasoning method which builds upon the theory of formal logic and axiomatic (analytic) theory of science
2. an informal reasoning method that builds upon psychology and cognition (J. H. Newman, in Reese, 1980)
3. an informal reasoning method that is based on the theory of informal logic (Blair, 1980)
4. a formal reasoning based on and justified by the theory of dialectical logic, also known as dialogic (Kamlah & Lorenzen, 1984)

Our current work involves formal reasoning of both the first and fourth kind and informal reasoning of the third kind. This paper addresses only the first kind of reasoning—i.e., reasoning in the traditional sense of formal reasoning, and deductive inference in particular. The basis of formal reasoning is logical inference. Inference in general can be deductive, inductive, or abductive. Formal reasoning can be both exact and inexact. Thus, there are exact and inexact rules of logical inference. Here, we consider only exact reasoning. For a formal inexact reasoning approach we have in mind the theory of Fuzzy sets and Fuzzy logic, which deals with inexact or approximate reasoning (Zadeh, 1965, 1979, 1983, and 1985).

In general, IBIS participants raise issues, take positions from the issues, and advance arguments supporting or objecting to the positions. The problem is resolved when the root issue and all other related (major) issues are resolved. Resolving issues involves evaluating (supporting and objecting) arguments to help us find, and thus select, *the most supported and the least objected-to positions*. What we have just said amounts to a *decision-procedure* that we can include in an IBIS-based *decision support system* (DSS). One way to represent such a decision procedure is to use the relational algebraic operation of *quotient*, which we can easily represent in Prolog.

If we have two entities A and B with a respective arity of j and k, expressed as $j > k$, then the quotient, denoted as $A \div B$ is a relation with the set of $(j-k)$ -tuples t such that:

$$A \div B = A \langle 1, 2, \dots, j-k \rangle \text{ -- } ((A \langle 1, 2, \dots, j-k \rangle ** B \text{ -- } A) \langle 1, 2, \dots, j-k \rangle$$

The double dash (--) and the double asterisk (**) represent set difference and cartesian product, respectively. To understand "quotient" without the effort of unfolding this complex formula let's use an example. If we have the following relations A and B:

A			
a1	a2	a3	a4
r	s	t	v
r	s	w	x
s	t	w	x
w	v	t	v
w	v	w	x
r	s	v	w

B	
b1	b2
t	v
w	x

these relations are given in Prolog as the following set of facts:

```
a(r,s,t,v).
a(r,s,w,x).
a(s,t,w,x).
a(w,v,t,v).
a(w,v,w,x).
a(r,s,v,w).
b(t,v).
b(w,x).
```

Then, the quotient expressed in Prolog (a modified version of the one in Li, 1984) is the relation:

```
quotient(A1,A2):-
    group([A1,A2],a(A1,A2,_),[A1,A2]),
    setof([AB1,AB2],a(A1,A2,AB1,AB2),Set2), /* built-in */
    setof([AB1,AB2],b(AB1,AB2),Set1),
    subset(Set1,Set2).
```

Li defines group as a "partitioning relation which conceptually rearranges the relation into groups such that in any one group all tuples have the same value for the grouped attribute."

Thus we can write the following definition:

```
:- dynamic ffound/1.
group(N,G,N):-
    call(G),
    only(N).
only(N) :-
    \+(ffound(N)),
    asserta(ffound(N)).
subset is defined as follows:
subset([H|T],S):-
    member(H,S),
    subset(T,S).
subset([],_).
```

Now, if we try the "quotient" goal, Prolog's response would be:

```
| ?- quotient(X,Y).
```



```

X = r
Y = s;
X = w
Y = v;
no
| ?-

```

Put in a relational form, the result is the relation `a%%b` with two tuples:

a%%b	
abl	ab2
r w	s v

To resolve issues in IBIS, we first need the following relations: `responds_to(P,I)`, `objects_to(A,P)`, `supports(A,P)`, `accepted(A)`, and `rejected(A)`. We have already discussed how to abstract the first three relations in the section on abstractions. Here are the definitions for “accepted” and “rejected”:

```

accepted(Aeid):-
    node(Aeid,AttrList),
    member(attr(acceptance-status,accepted),AttrList).
rejected(Aeid):-
    node(Aeid,AttrList),
    member(attr(acceptance-status,rejected),AttrList).

```

We also define the quotient relations `supports%%accepted`, `supports%%rejected`, `objects_to%%accepted`, `objects_to%%rejected` in the following form:

```

/* positions supported by accepted arguments */
su_quotient_ac(P):-
    retractall(ffound(_)),
    group([P],supports(A,P),[P]),
    setof([A],supports(A,P),Set2),
    setof([A],accepted(A),Set1),
    subset(Set1,Set2).

/* positions supported by rejected arguments */
su_quotient_re(P):-
    retractall(ffound(_)),
    group([P],supports(A,P),[P]),
    setof([A],supports(A,P),Set2),
    setof([A],rejected(A),Set1),
    subset(Set1,Set2).

/* positions objected-to by accepted arguments */
ob_quotient_ac(P):-
    retractall(ffound(_)),
    group([P],objects_to(A,P),[P]),
    setof([A],objects_to(A,P),Set2),
    setof([A],accepted(A),Set1),
    subset(Set1,Set2).

/* positions objected-to by rejected arguments */
ob_quotient_re(P):-
    retractall(ffound(_)),
    group([P],objects_to(A,P),[P]),
    setof([A],objects_to(A,P),Set2),
    setof([A],rejected(A),Set1),
    subset(Set1,Set2).

```

Now we can use these definitions as constraints on selecting a position. A definition that cap-

tures such constrained decision making is the following:

```
selected(P):-
    su_quotient_ac(P),
    \+(su_quotient_re(P)), /* \+ is Quintus-prolog's "not" */
    \+(ob_quotient_ac(P)),
    ob_quotient_re(P).
```

This definition is stated in English as follows:

A position P could be (possibly) selected IF
it has accepted supporting arguments AND
none of its supporting arguments are rejected AND
none of the arguments objecting to it was accepted AND
it has rejected arguments objecting to it.

To try out this definition, we give the following goal:

```
| ?- selected(P).
P = p1;
no
| ?-
```

We can take this definition one step further by considering the possible (or near) resolution of an issue if that issue has at least one selected position:

```
resolved(I):-
    responds_to(P,I),
    selected(P).
```

The above-mentioned decision procedure is only part of an IBIS-based expert system prototype for systems design and analysis. Another part of the system is the IBIS-etiquette adviser shown in Figure 3.

EXAMPLE 2: An Expert System for Implementing the QFD Methods

The *simultaneous engineering* research project at Rockwell International (an MCC shareholder) is an effort to develop tools for supporting the integrated product development process. Simultaneous engineering (SE) is also known as *concurrent engineering* or *integrated product development*. The goal of SE is to model a product development process that results in higher quality and lower cost and that requires shorter time to market than traditional product development systems.

In SE, the different (independent or related) processes of planning, design, manufacturing, testing, and in-service are considered in parallel. The traditional (non-simultaneous) systems engineering approach tackles the different sub-processes sequentially. In many ways, the sequential engineering process has been found to be the main reason for the increase in engineering change orders, the increase in design cycle time, the high manufacturing costs, the increase in scrap and rework situations, and the unnecessary complexity and bad quality of the final product.

The task of SE is to automate the management of planning-to-production processes, taking into consideration the concurrences and cross-functionality of the different processes. Thus, it deals with more than one or two categories or fields of knowledge and expertise. This implies that SE needs more than one method, technology, or instrument to achieve a particular end. These methods or technologies can be alternative, complementary, or independent. In general, we believe that any SE system should allow us to coordinate the competing, or complementing, or interacting methodologies or subsystems.

Quality Functional Deployment (QFD), also known as the *House of Quality* method, is an approach developed by the Japanese to help coordinate the integrated product development pro-

cess (see Hauser, 1988 and Eureka, 1988). The QFD method seeks to diffuse customer-desired qualities (attributes) into a product through the design, specification, parts deployment, process planning, and production planning stages.

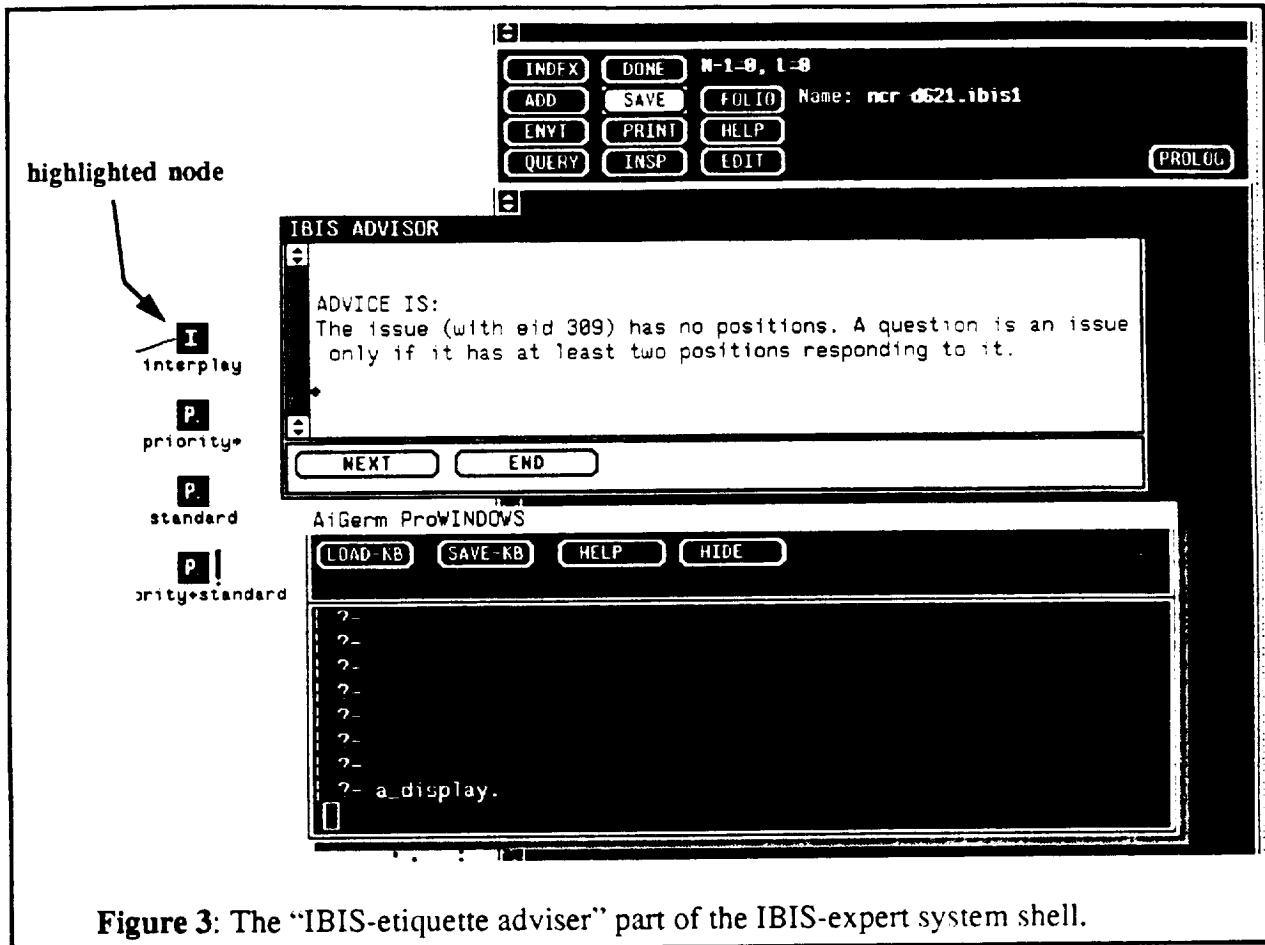
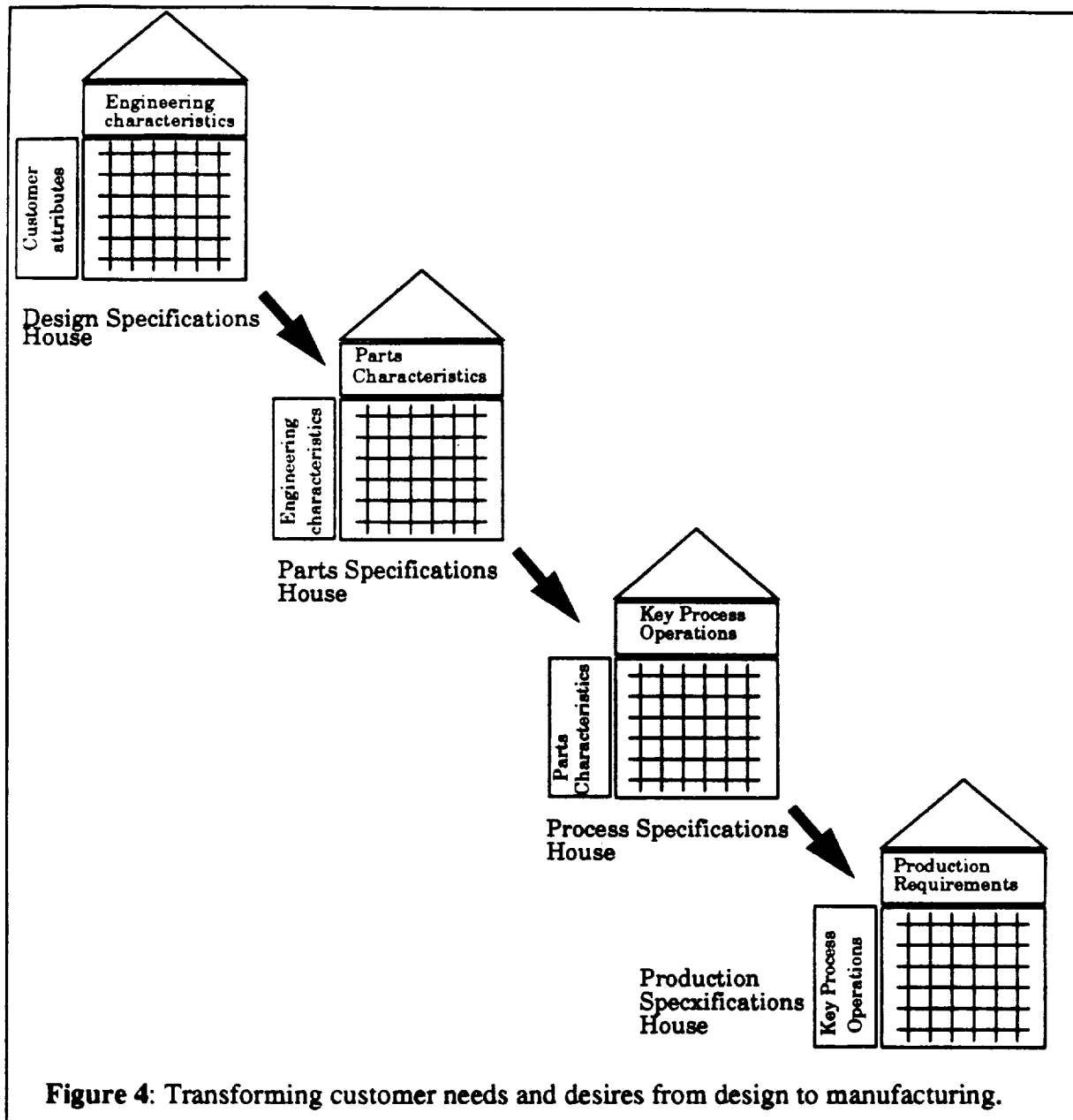


Figure 3: The "IBIS-etiquette adviser" part of the IBIS-expert system shell.

Thus, QFD could serve as a general (and integral) structuring and coordinating part of the SE process. Traditionally, QFD is implemented using linked houses, see Figure 4, with each house being a matrix for relating qualities that convey the customer's voice through to manufacturing. In our case, we want to automate the house-building process and provide decision support for resolving the customer-needs satisfaction issues. One way to look at QFD is to view it as a problem solving process involving a group of participants with different backgrounds—i.e. customers, designers, manufacturing engineers, marketing people, managers, and so on—engaged in a series of discussions trying to resolve different issues. Once we accept such a view, we are tempted to use the IBIS method to represent the QFD-group interactions.

Using AiGerm, we wrote an IBIS-based QFD expert system to help automate the construction of QFD houses. For example, in the case of the first house, the system would elicit needs from customers and help in deriving the engineering characteristics required in the design specifications. Figure 5 shows the network generated in cooperation between the customer and the QFD-expert rules of the system.



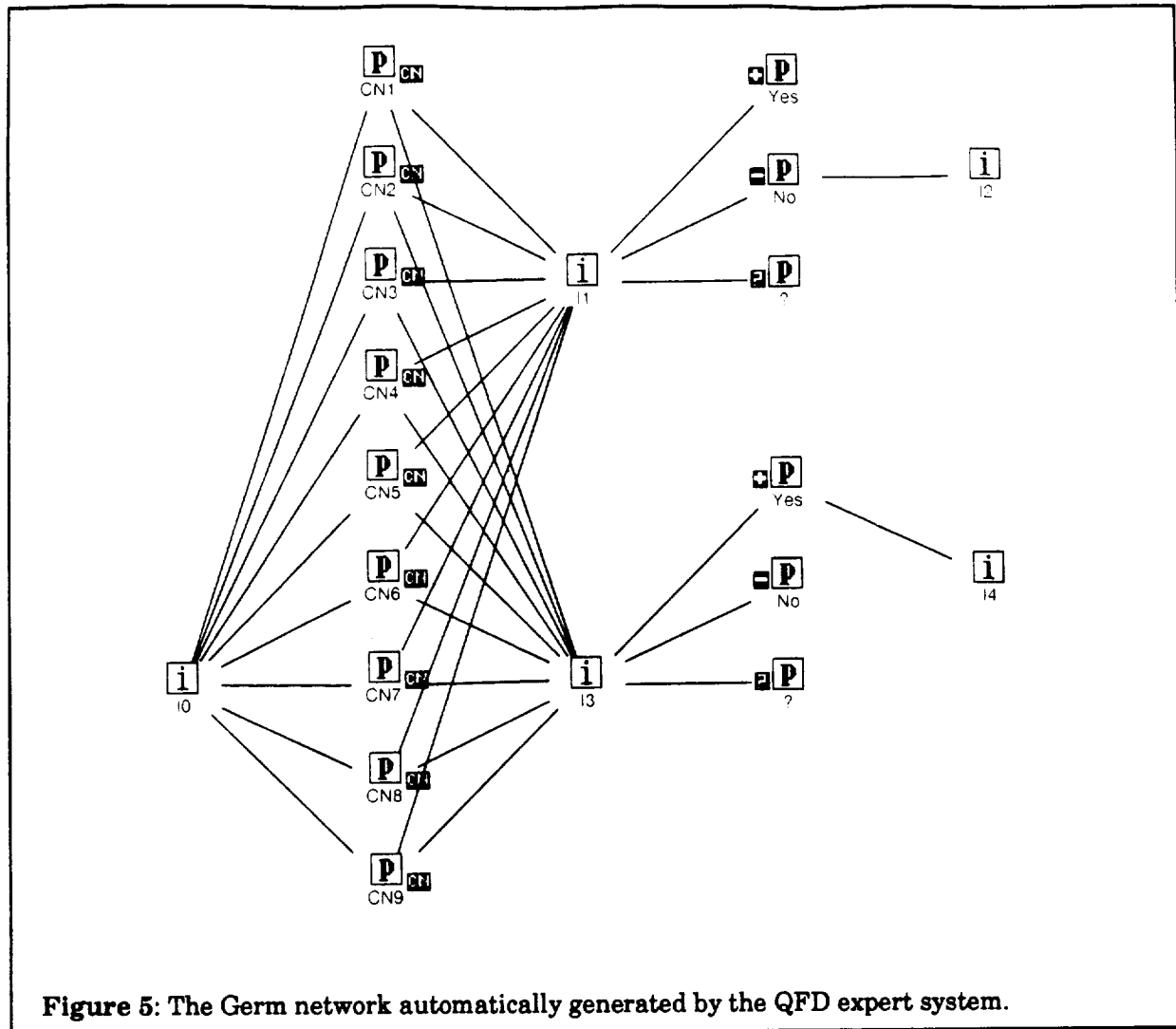


Figure 5: The Germ network automatically generated by the QFD expert system.

Conclusion

AiGerm is MCC's Germ with a logic programming front end. It treats a Germ network as a knowledge base made up of node, link, and aggregate base relations. Users of AiGerm can use Prolog, or MCC's LDL either to navigate Germ networks through queries or to develop prototypes of knowledge-based hypermedia systems. For both applications, we have found that abstractions are the necessary building blocks for any serious use of the system. Currently, AiGerm is used in two major applications, MCC's software design information recovery tool (DESIRE), and Rockwell International's Simultaneous Engineering research project. In conclusion we believe that AiGerm is a cost effective tool for developing and testing systems design prototypes.

Acknowledgments

The author would like to thank Frank Wrabel and the Simultaneous Engineering research team for their many insightful observations during conversations about the research being done at the Space Systems Division of Rockwell International. The author also wishes to thank Noreen Garrison, of STP, for her valuable help in editing the paper.

References

- [Blair, 1980] Blair, J. A. and R. H. Johnson. *"Informal Logic: The First International Symposium."* Inverness, California: Edgepress, 1980.
- [Biggerstaff, 1988] Biggerstaff, T. J. *"Design Recovery for Maintenance and Reuse."* MCC Technical Report, STP-378-88, November, 1988.
- [Bratko, 1986] Bratko, I. *PROLOG Programming for Artificial Intelligence.* Wokingham, England: Addison-Wesley, 1986.
- [Clocksin, 1981] Clocksin, W. and C. S. Mellish. *Programming in PROLOG.* Berlin: Springer Verlag.
- [Conklin, 1988] Conklin, J. and M. L. Begeman (1988). "gIBIS: A Hypertext Tool for Exploratory Policy Discussion." *ACM Transactions on Office Information Systems*, 6 (October 1988), pp. 303-331.
- [Consens, 1989] Consens, M. P. and A. O. Mendelzon. "Expressing Structural Hypertext Queries in GraphLog." *Hypertext '89 Proceedings*, November 5-8, 1989, Pittsburgh, Pennsylvania, pp.269-292.
- [Eureka, 1988] Eureka, William E. and N. E. Ryan. *"The Customer Driven Company: Managerial Perspectives on QFD."* Dearborn, Michigan: ASI Press, 1988.
- [Halasz, 1989] Halasz F. and J. Conklin. *"Issues in the Design and Application of Hypermedia Systems."* SIGCHI 89, 1989.
- [Hashim, 1990a] Hashim, S. H. *"MicroIBIS: A Micro Issue Based Information System."* In: Exploring Hypertext Programming: Writing Knowledge Representation and Problem Solving Programs, Part III." Blue Ridge Summit, PA: Windcrest Books, imprint of TAB BOOKS, 1990.
- [Hashim, 1990b] Hashim, S. H. *"WHAT: Writing with a Hypertext-based Argumentative Tool."* MCC/STP Technical report, No. STP-270-90, 1990.
- [Hashim, 1990c] Hashim, S. H. and Mahesh Zurale. *"AiGerm: An Intelligent Graphical Entity relational Modeler."* MCC/STP Technical Report, No. STP-096-90, 1990.
- [Hauser, 1988] Hauser, J. R. "The House of Quality." *Harvard Business Review*, May-June, 1988.
- [Kamlah, 1984] Kamlah, W. and P. Lorenzen. *Logical Propaedeutic: Pre-School of Reasonable Discourse.* Lanham, MD: University Press of America, Inc., 1984.
- [Kunz, 1970] Kunz, W. / Rittel, H. W.J. *"Issues as Elements of Information Systems."* Institute for Urban and Regional Development, University of California, Berkeley, No.131, 1970; also: Institut fuer Grundlagen der Planung, Universiteat Stuttgart S-78-2.
- [Li, 1984] Li D. *A Prolog Database System.* England: Research Studies Press, 1984.
- [Reese, 1980] Reese, W.L. *Dictionary of Philosophy and Religion, Eastern and Western Thought.* Atlantic Highlands, N.J.: Humanties Press Inc, 1980.

- [Rittel, 1980] Rittel, H. W. J. "APIS - A Concept for an Argumentative Planning Information System." Institute of Urban and Regional Development, University of California, Berkeley, Working Paper 324, 1980. Also Institut fuer Grundlagen der Planung, Universiteat Stuttgart S-80-2.
- [Zadeh, 1980] Zadeh, L. A.. "Inference in Fuzzy Logic." *Proceedings of the International Symposium on Multiple-Valued Logic*, Northwestern University (1980), pp. 124-131.
- [Zadeh, 1979] Zadeh, L. A.. "A Theory of Approximate Reasoning." *Machine Intelligence*, ed. D. Michie, American Elsevier, 1979, pp.149-194 .
- [Zadeh, 1965] Zadeh, L. A. "Fuzzy Sets." *Information and Control*, 8 (1965), pp.338-353.

N91-21958

HEAVENS System for Software Artifacts

Paul Matthews

Bellcore Company

The HEAVENS system is a workstation-based collection of software for analyzing, organizing and viewing software artifacts. As a prototype, the system has been used for visualizing source code structure, analyzing dependencies, and restructuring to simplify maintenance. The system has also been used in the early stages of software design to organize and relate design objects, maintain design documentation, and provide a ready-made framework for later coding.

HEAVENS System for Software Artifacts

PAUL MATTHEWS

Bellcore

Computer Technology Transfer Division

Piscataway, New Jersey 08854

paul@ctt.bellcore.com

908-699-7461

Hypermedia '90

Houston, Texas

December 1990

HEAVENS - IN THE BEGINNING

Circa 1987

Lots of software engineering documents.

Documents are hard to relate.

Useful information is buried and hard to find.

+

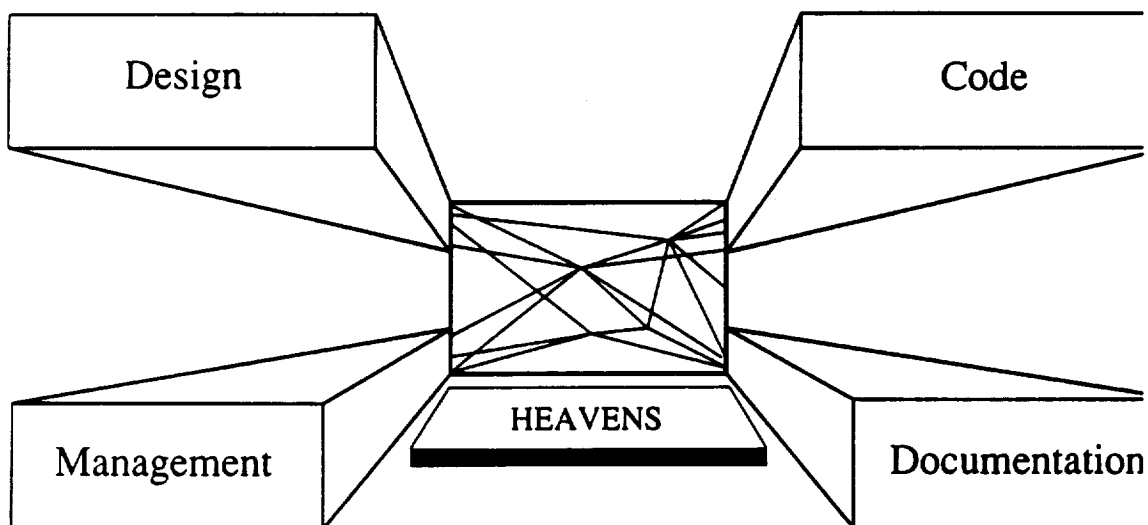
Linking document parts.

Workstation user interface.

=

Hypertext strategy.

HEAVENS - BRIDGING THE GAPS



HEAVENS - SOFTWARE ARTIFACTS

Characteristics

Software is relatively highly structured.

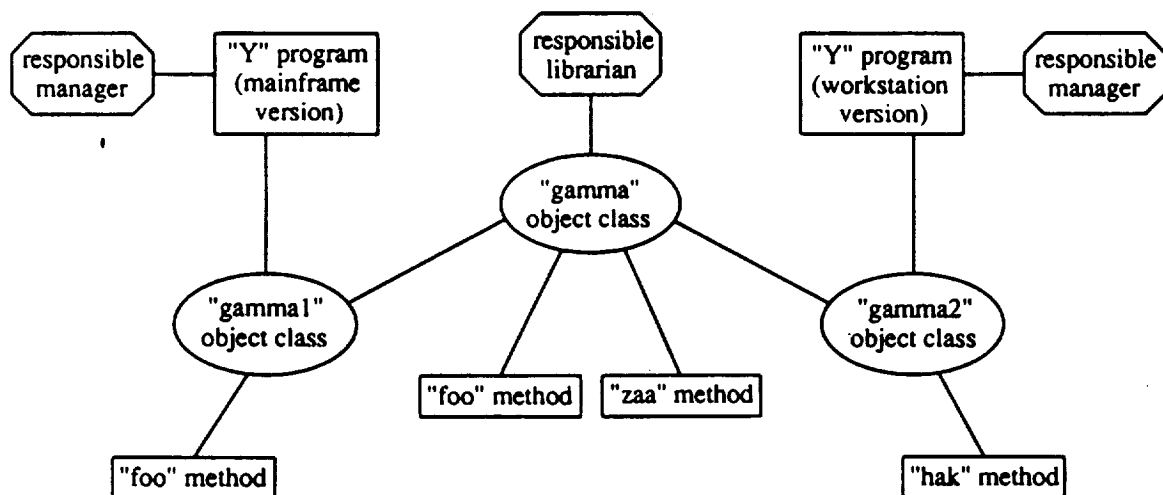
Structure itself is a managed asset.

Extension and maintenance are significant.

Software is manipulated by power tools.

PCM
Boltman
12/90

HEAVENS - SOFTWARE ARTIFACTS



PCM
Boltman
12/90

HEAVENS - SOFTWARE ARTIFACTS

Basic Hypertext Requirements

Classify nodes and links.

Visualization.

Power tool attachments.

FCM
Release
12/90

HEAVENS - MCC CONNECTION

GERM

Provides many useful facilities:

Node-and-link graphics.

Text associated with graphical elements.

External node-and-link schemata.

Plug-in modules.

FCM
Release
12/90

HEAVENS - SOFTWARE ARTIFACTS

Chosen for Study

Initially:

C language source code.

Program configuration.

Later:

Object-oriented design.

PCM
Belmont
12/90

HEAVENS - C LANGUAGE

Features

Import / export.

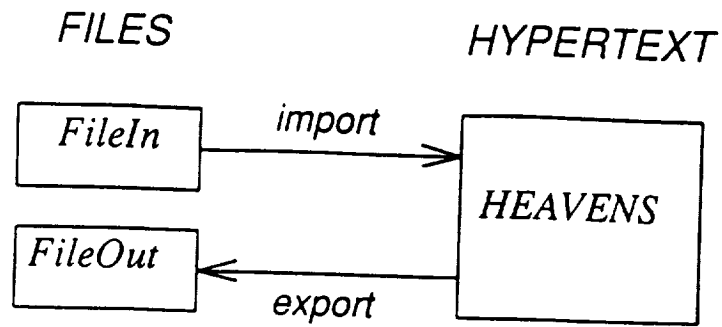
Dependency analysis.

Automatic "make".

PCM
Belmont
12/90

HEAVENS - C LANGUAGE

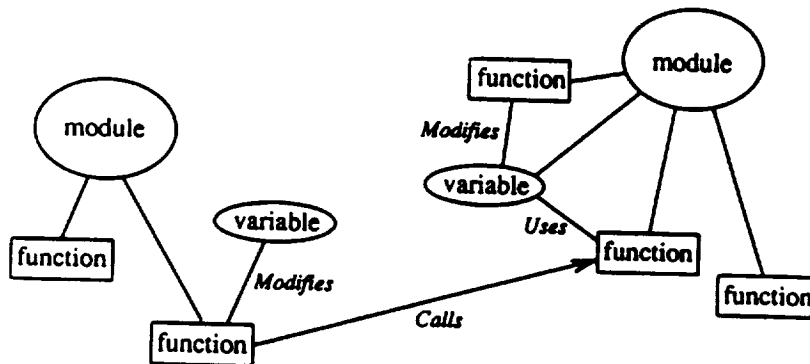
Import / Export



POM
Bottom
12/90

HEAVENS - C LANGUAGE

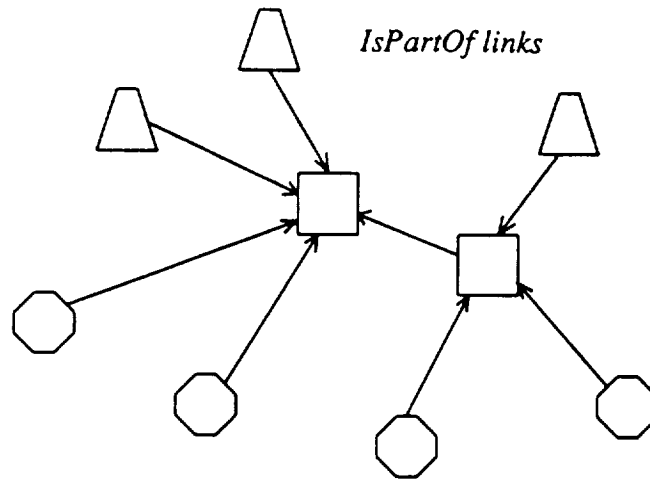
Dependency Analysis



POM
Bottom
12/90

HEAVENS - C LANGUAGE

Automatic "Make"



POM
Belmont
12/90

HEAVENS - O-O DESIGN

Purpose

Record design decisions.

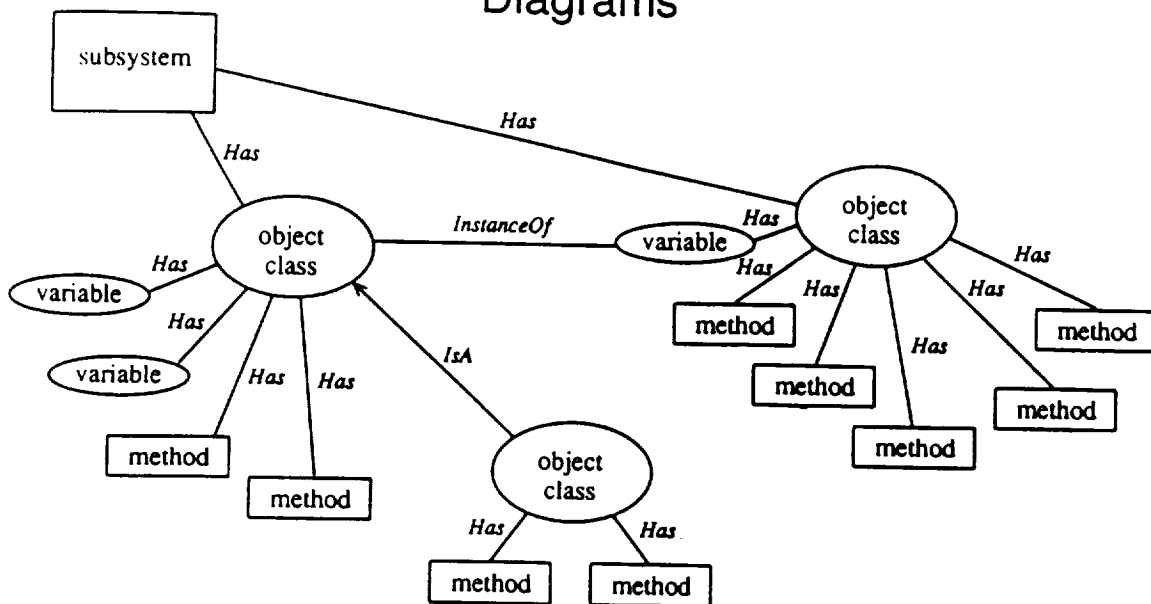
Document object classes.

Framework for coding.

POM
Belmont
12/90

HEAVENS - O-O DESIGN

Diagrams



PGM
Bottom
12/90

HEAVENS - O-O DESIGN

Class Description Documents

Class name:

SuperClasses:

SubClasses:

General Description:

Member Variables:

Member Functions:

...

PGM
Bottom
12/90

HEAVENS - HOW IT IS USED

Best Selling Features

C Language:

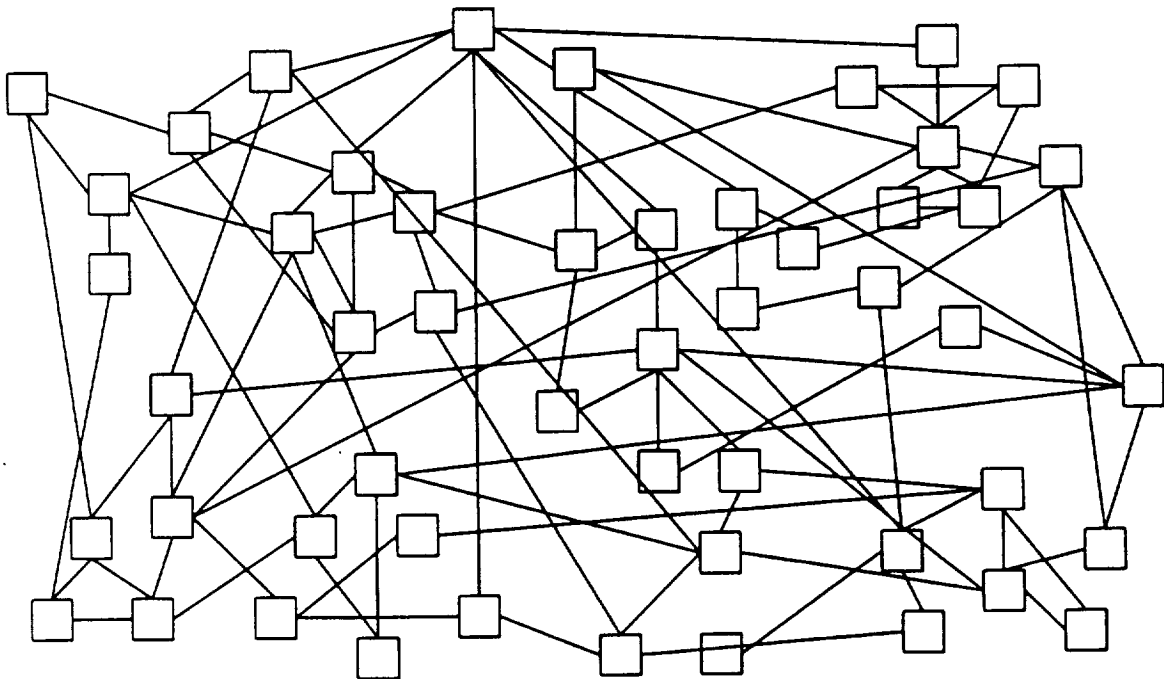
Complexity assessment.

OO Design:

Class documentation.

POM
Bellman
12/90

HEAVENS - DISPLAY OVERLOAD



POM
Bellman
12/90

HEAVENS - INTELLIGENT DISPLAY

Logical "Reasoning"

Specifying complex operations.

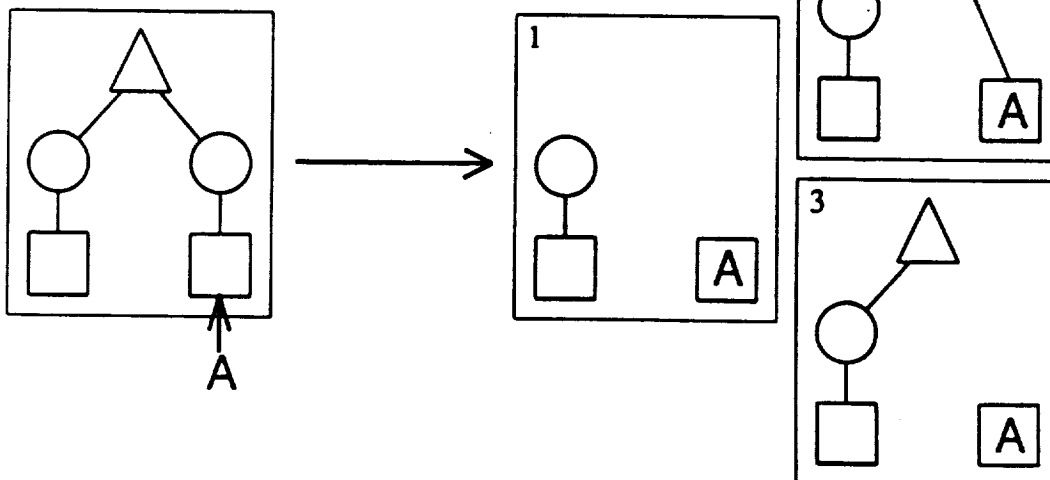
Query processing.

Deriving new nodes and links.

PGM
Belmont
12/90

HEAVENS - INTELLIGENT DISPLAY

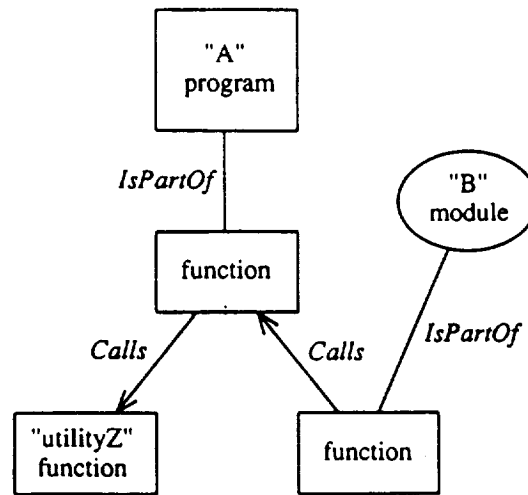
e.g., Aggregation



PGM
Belmont
12/90

HEAVENS - INTELLIGENT DISPLAY

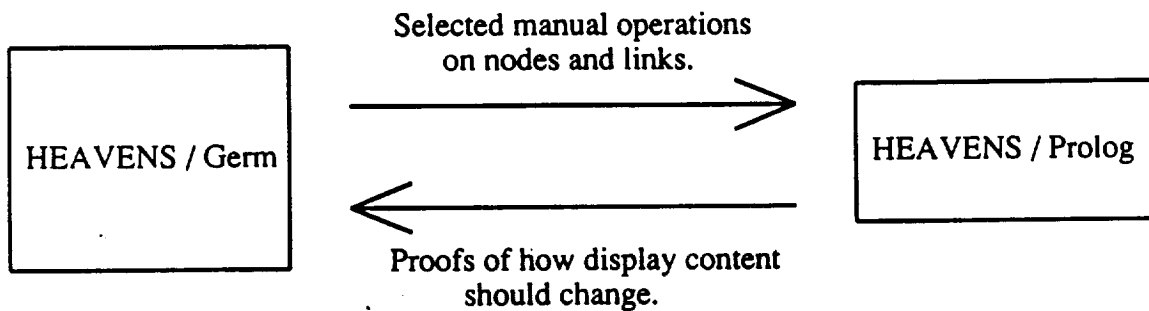
Pattern Query



POM
Bellcore
12/90

HEAVENS - INTELLIGENT DISPLAY

Prolog Experiments



POM
Bellcore
12/90

HEAVENS - SOFTWARE ARCHITECTURE

Drivers

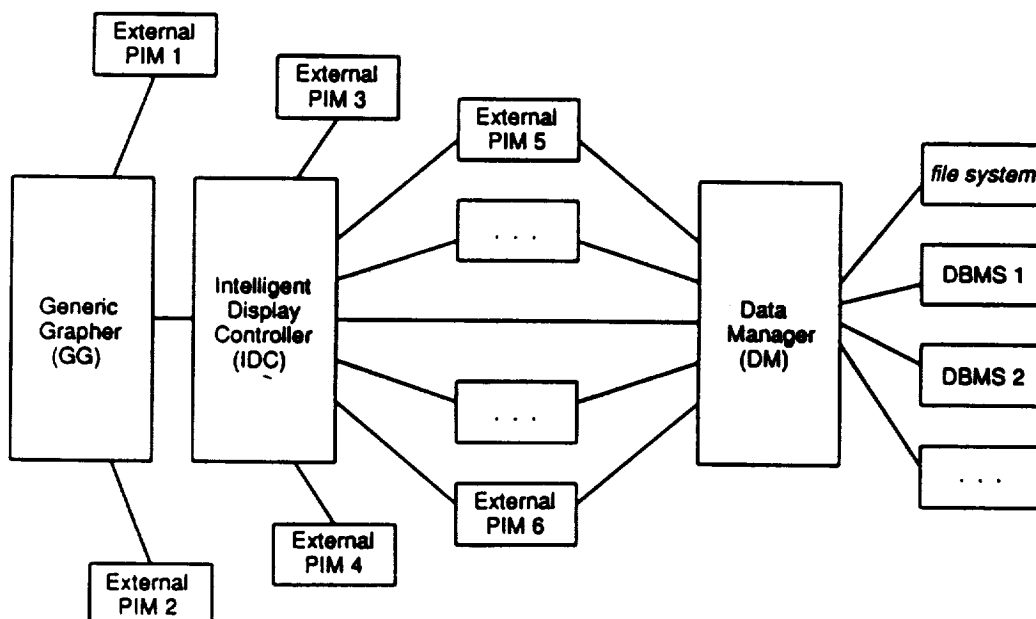
Multiple hardware and software platforms.

Extending functionality.

Integration with other components.

PGM
Software
12/90

HEAVENS - SOFTWARE ARCHITECTURE



PGM
Software
12/90

HEAVENS - SUMMARY

Fine granularity.

Types.

Tool attachments.

Logical reasoning.

Architecture for extensibility.

POM
Bellman
12/90

HEAVENS - FUTURE

Busy Person Environment

Clients demand ever increasing productivity.

Software workers perform more diverse tasks.

Tasks require more complex information.

Doing it the (old) "right way" may not be humanly possible.

POM
Bellman
12/90

HEAVENS - FUTURE

Workstation for the Busy Person

Hot-spot identification.

Signal what is urgent.

Advice on resources.

Intelligent automation.

Collaboration support.

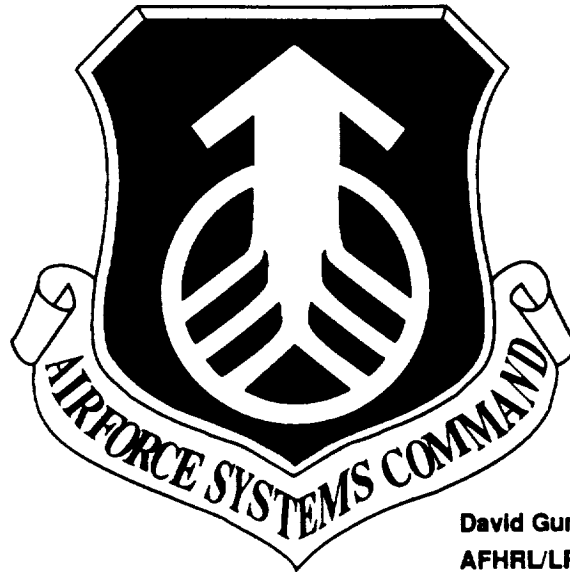


Panel: Three Issues for Real-World Hypertext Projects

David Gunning

Human Resources Laboratory, Wright-Patterson AFB, OH

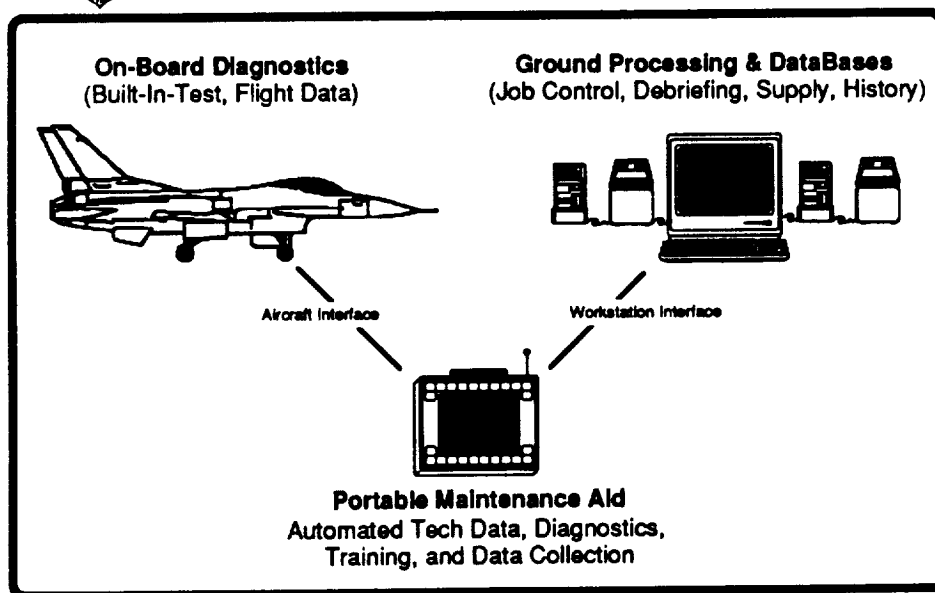
INTEGRATED MAINTENANCE INFORMATION SYSTEM



David Gunning
AFHRL/LRC
Wright-Patterson AFB, OH

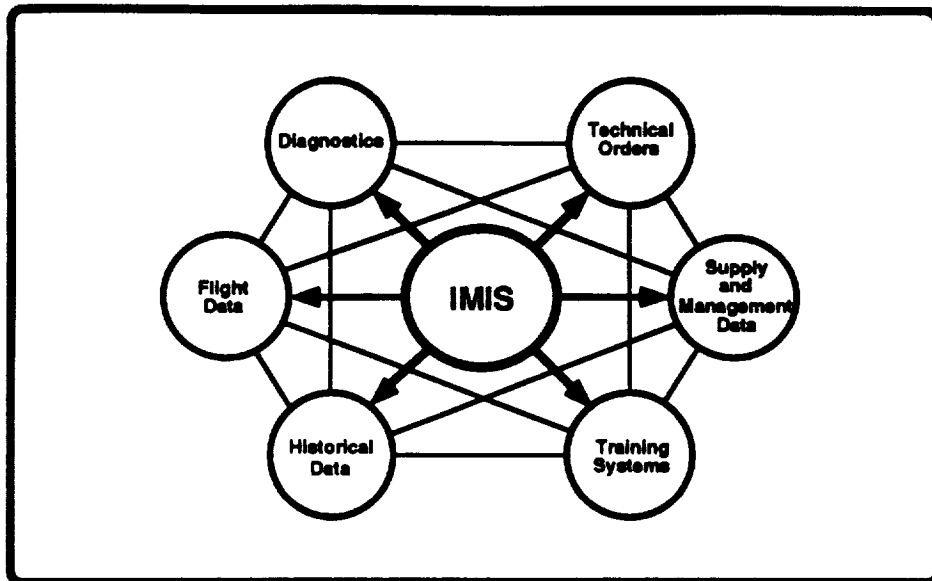


IMIS - A CALS CONCEPT FOR MAINTENANCE

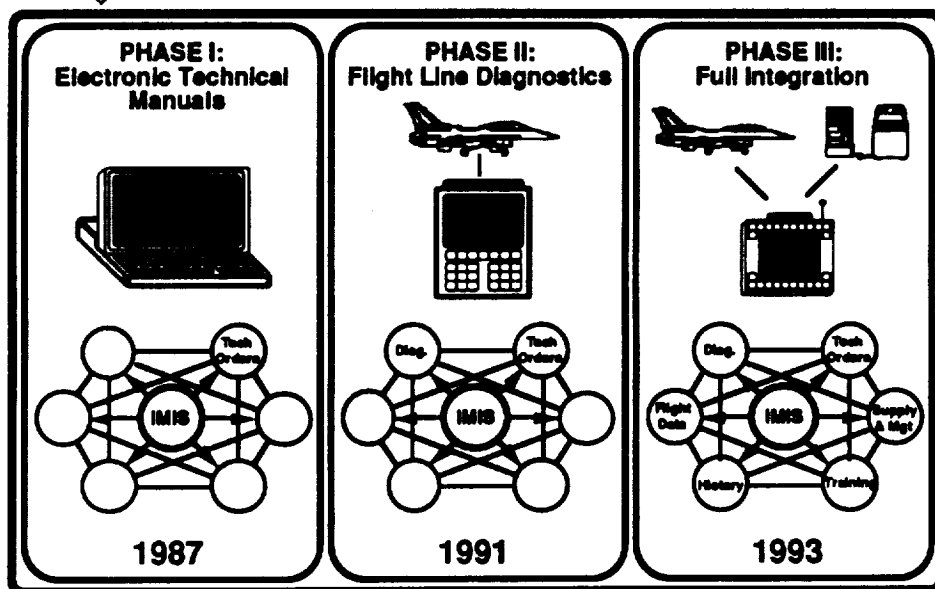




INFORMATION INTEGRATION



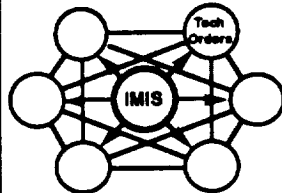
PHASED APPROACH





IMIS PHASE I: ELECTRONIC PRESENTATION

PHASE I: Electronic Technical Manuals



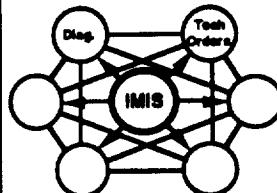
1987

- **Electronic Presentation System**
 - Off-The-Shelf Computer
 - Presentation Formats for Interactive Display
- **In-Shop Field Tests**
 - Initial Test at Offutt AFB 1984
 - Retest at Grissom AFB 1985
 - Independent Navy Test 1986
- **Performance Results**
 - 100% FI Success vs. 75% with Paper
 - Average Fault Isolation Time Halved
 - No False Removals
 - Technicians Preferred Electronic System



IMIS PHASE II: FLIGHT LINE DIAGNOSTICS

PHASE II: Flight Line Diagnostics

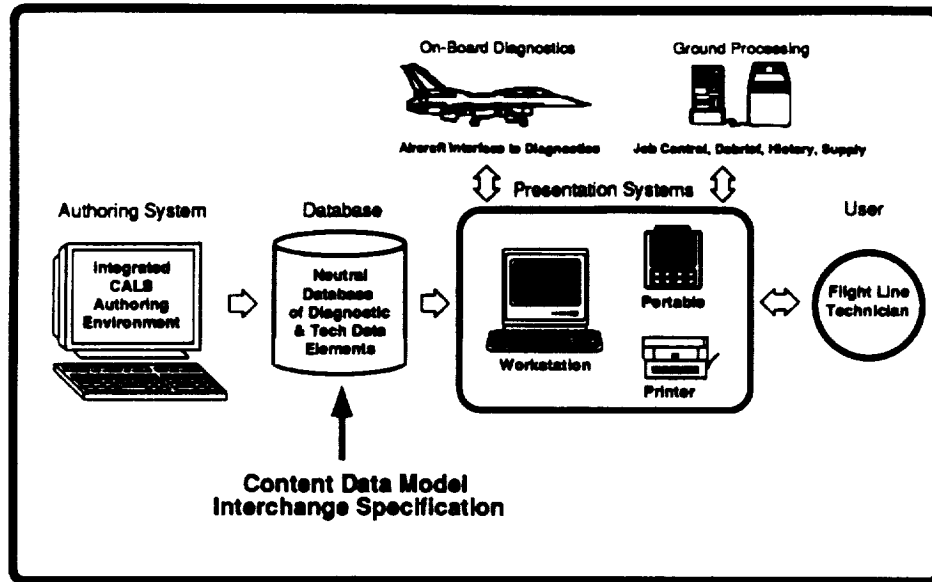


1991

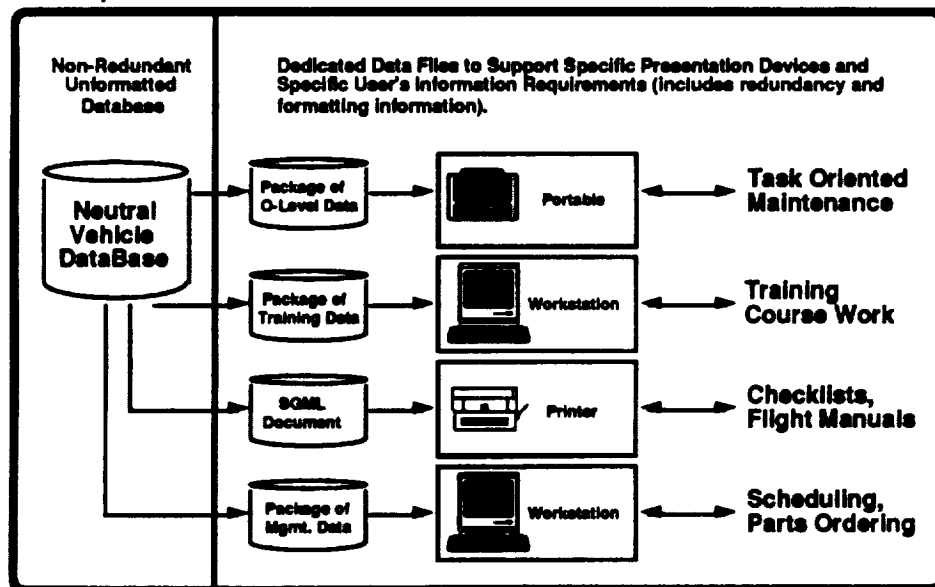
- **Flight Line Diagnostic System**
 - Portable Maintenance Aid (PMA)
 - 1553 Aircraft Interface
 - Integrated Diagnostics / Tech Data Software
- **Authoring System for "Type C" Data**
- **Flight Line Field Tests**
 - Initial PMA Demo at MacDill AFB 1988
 - F-16 Test at Homestead AFB 1989
 - F/A-18 Test at Patuxent River 1991
 - F/A-18 Test at Cecil Field 1991
- **Content Data Model (CDM)**
- **Tri-Service Specification**



CONTENT DATA MODEL



USING NEUTRAL, INTEGRATED DATA



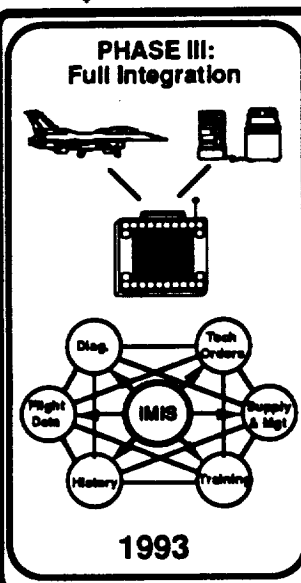


Interactive Electronic Technical Manuals (IETM) Specifications

- **General Content, Style, Format, and User Interaction**
General specification of the technical content, writing style, display formatting, and user interaction requirements for an IETM system.
- **Revisable Data Base (Content Data Model)**
Detailed specification of the data model (i.e., data entities, attributes, and relationships) required for a neutral database of IETM information.
- **Quality Assurance**
General specification of the quality assurance steps necessary for developing, validating, and verifying IETM data.
- **View Package Handbook**



IMIS PHASE III: FULL INTEGRATION

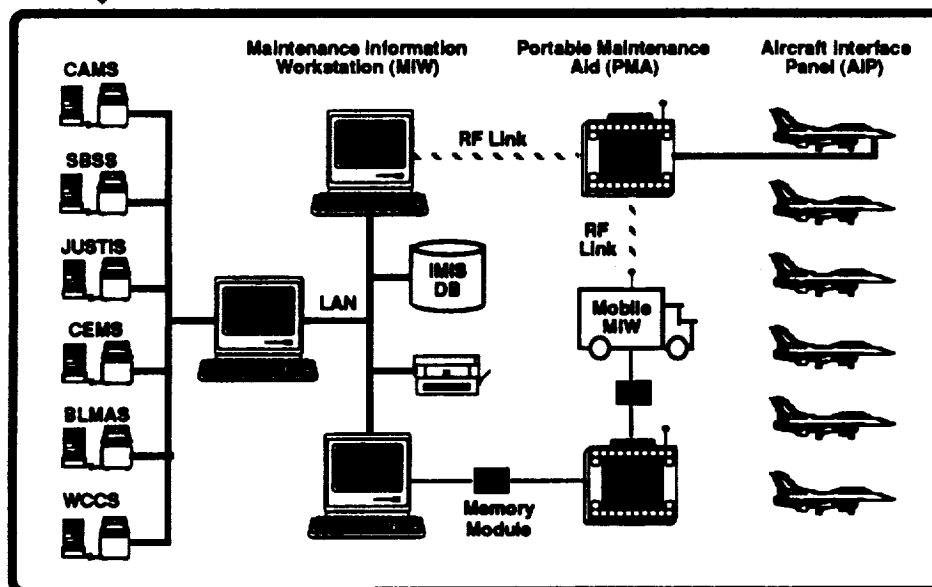


- **Fully Integrated Information System**
 - PMAs + Maintenance Workstations
 - Interfaces with A/C, CAMS, SBSS, etc.
 - Information Integration Software
- **Detailed Requirements Analysis**
 - Activity Model of Maintenance Functions
 - Data Model of Information Elements
 - Computer Systems Architecture
- **Base-Level Field Test of IMIS Prototype**
- **Concept Applications Specifications**

Activity Model of O-Level Maintenance



IMIS ARCHITECTURE





IMIS APPLICATIONS

- **Phase I: Electronic Technical Manuals**
 - B-2 Improved Technical Data System (ITDS)
- **Phase II: Interactive Diagnostics**
 - JSTARS Computerized Technical Order System (CTOS)
 - F-16 "Type C" Retrofit
 - Army Contact Test Set for the M-1 Tank
- **Phase III: Fully Integrated System**
 - ATF Integrated Maintenance System (AIMS)
 - A-12 Interactive Electronic Technical Manual System



Software Engineering Professional Education Center

University of Houston-Clear Lake

2700 Bay Area Blvd., Box 258

Houston, Texas 77058

(713) 282-2223